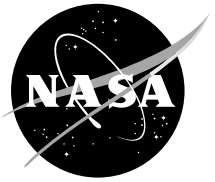
NASA/TP—2000–209902



Comprehensive Design Reliability Activities for Aerospace Propulsion Systems

R.L. Christenson and M.R. Whitley
Marshall Space Flight Center, Marshall Space Flight Center, Alabama

K.C. Knight
Sverdrup Technology, Huntsville, Alabama

January 2000

The NASA STI Program Office…in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and mission, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results…even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at ***http://www.sti.nasa.gov***

- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at (301) 621–0134

- Telephone the NASA Access Help Desk at (301) 621–0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076–1320

NASA/TP—2000–209902

Comprehensive Design Reliability Activities for Aerospace Propulsion Systems

R.L. Christenson and M.R. Whitley
Marshall Space Flight Center, Marshall Space Flight Center, Alabama

K.C. Knight
Sverdrup Technology, Huntsville, Alabama

National Aeronautics and
Space Administration

Marshall Space Flight Center • MSFC, Alabama 35812

January 2000

Acknowledgments

The authors would like to thank the following who made important contributions directly and indirectly to this effort: Charles Pierce, Richard Ryan, Brenda Lindley-Anderson, David Seymour, and Tom Byrd. A special thanks to Sid Lishman who supported the extensive analyses needed to support the special reliability topics and the quality data discussion.

Available from:

NASA Center for AeroSpace Information
7121 Standard Drive
Hanover, MD 21076–1320
(301) 621–0390

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
(703) 487–4650

TABLE OF CONTENTS

TABLE CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF ACRONYMS

ARAM	automated reliability/availability/maintainability
AQL	acceptable quality level
ASME	American Society of Mechanical Engineers
ASSIST	abstract semi-Markov specification interface
BMOD	bill of material object damage
C/O	cutoff
Ca phen	carbon phenolic
CARE III	computer-aided reliability estimation, third generation
CDR	critical design review
CEI	contract end item
CW	critical items list
DDT&E	design, development, test, and evaluation
DoD	Department of Defense
disassy	disassembly
E&M	electrical and mechanical
EFD	equivalent full duration
EMA	electro-mechanical actuator
ETARA	event time availability, reliability analysis
FEAS–M	failure environment analysis system at MSFC
FEAT	failure environment analysis tool
FMEA	failure modes and effects analysis
FMECA	failure modes, effects, and criticality analysis
FTA	fault-tree analysis
GH_2	gaseous hydrogen
GHe	gaseous helium
GLOW	gross lift-off weight
gox	gaseous oxygen
HCF	high-cycle fatigue
He	helium
IEEE	Institute of Electrical and Electronics Engineers
IPS	interpropellant seal
Isp	specific impulse
LaRC	Langley Research Center
LCF	low-cycle fatigue
LH_2	liquid hydrogen
LN_2	liquid nitrogen
lox	liquid oxygen
LPFTP	low pressure fuel turbopump
MPS	main propulsion system

LIST OF ACRONYMS (Continued)

MSFC	Marshall Space Flight Center
MTBF	mean time between failure
MTBM	mean time between maintenance
MTTF	mean time to failure
MTTR	mean time to repair
NASA	National Aeronautics and Space Administration
NESSUS	numerical evaluation of stochastic structures under stress
NLS	National Launch System
NPRD	nonelectronic parts reliability database
PAWS	pade approximation with scaling
PDA	probabilistic design analysis
PDR	preliminary design review
PRA	probabilistic risk assessment
PRACA	problem reporting and corrective action
QA	quality assurance
QC	quality control
R&D	research and development
RBD	reliability block diagram
RCS	reaction control system
RELAV	reliability/availability
RID	review item disposition
RLV	reusable launch vehicle
rpm	revolutions per minute
S&MA	safety and mission assurance
SAIC	Science Applications International Corporation
SF	safety factor
SIRA	shuttle integrated risk assessment
Si phen	silica phenolic
SRM	solid rocket motor
SSME	Space Shuttle main engine
SSPRA	Space Shuttle probabilistic risk assessment
STEM	scaled taylor exponential matrix
STS	Space Transportation System
SURE	semi-Markov unreliability range evaluator
SV	servo-valve
TP	technical publication
TPS	thermal protection system
TQM	total quality management
UCR	unsatisfactory condition report

NOMENCLATURE

C_{ss}	coefficient of standard deviations
CV_o	coefficient of variation
E	contingency factor (%)
P	probability
Pc	chamber pressure
R	reliability
Z	safety index

TECHNICAL PUBLICATION

COMPREHENSIVE DESIGN RELIABILITY ACTIVITIES FOR AEROSPACE PROPULSION SYSTEMS

1. INTRODUCTION

Design is often described as the integration of art and science. As such, it is thought of as more of a "soft science" where the emphasis is on concepts and where early contradictions may require less precise approaches to problem solving. It is important to distinguish between this "conceptual" design and the process of design engineering. Design is the process associated with establishing options based on need and customer requirements. Design engineering is the process of conducting a design once a general set of requirements is in place. It is the latter that is of interest in this report.

Several good references[1–3] provide traditional definitions and extensively discuss the important attributes of mechanical design. Of key interest here is the process of design engineering. From Ryan and Verderaime: "…, the design process is the informal practice of achieving the design project requirements throughout all design phases of the system engineering process."[4] Also, McCarty states: "..., design is a process of synthesis and tradeoffs to meet a required set of functional needs (absolute criteria) within a set of allocated resources (variable criteria)."[5]

It follows that designing for reliability is also a process—a systems engineering process that supports design trades and decisions from a reliability perspective. This reliability perspective is acquired through the analysis of the design in "failure space." Like other systems engineering discipline analyses, this analysis should be as rigorous and quantitative as possible and must support each phase of the design with appropriate and increasing detail. It is critical to start this process early. It has been estimated that more than *85* percent of the life-cycle cost is determined by decisions made during conceptual and preliminary design.

The overriding concern in this technical publication (TP) is with propulsion systems' reliability and its impact on design. Several analyses have shown the predominance of propulsion system failures relative to other vehicle system failures.[6–8] Obviously, propulsion systems' reliability is a key factor in determining crew safety for manned vehicles. Estimates of the cost of failure of STS–51L range from $4.5 billion for direct costs to $7 billion if indirect costs are included, and a program delay of ≈3 yr. With a demand for higher levels of vehicle reliability and manned vehicle safety, the need for comprehensive design reliability activities in all design phases has grown. Also, the need for an approach to track reliability throughout all phases of design and development activity has grown. Reliability improvements must be given higher priority for next-generation launch vehicles.

The need for understanding potential design failures supports another design perspective. "The purpose of design is to obviate failure."[2] The ability of a design to lessen the risk of failure may be

constrained due to the inherent difficulties in satisfying design requirements. Pye expresses it well: "The requirements for design conflict and cannot be reconciled. All designs for devices are in some degree failures, either because they flout one or another of the requirements or because they are compromises, and compromise implies a degree of failure."[1] It is therefore critical that timely and accurate reliability information be provided the designer throughout the design process. Thus, the case is made again that reliability is the first-order concern for any launch vehicle. The cost of unreliability, with its resulting loss of payload, loss of service, and extended repair time, makes failure prohibitive. Good design reliability engineering with good reliability estimation techniques and reliability models is required of an overall launch vehicle design strategy to ensure reliability.

Any new space launch vehicle system must significantly reduce the cost of access and payload to orbit to be economically viable in either the Government or commercial sectors. In addition, both developmental and operational risk must be maintained or improved. This is reflected in the current joint industry-Government X–34, X–33, and reusable launch vehicle (RLV) programs. In order to achieve significant reductions in program cost while maintaining acceptable risk, detail trades must be conducted between all other system performance parameters. Thus, cost and risk become design parameters of equal importance to the classical performance parameters, such as thrust, weight, and specific impulse (Isp).

Reliability is a major driver of both cost and risk. The results of reliability analyses are direct inputs to cost and risk analyses. Cost is also heavily driven by operations,[9] which also receives direct inputs from reliability analyses. As implied, cost and risk, and thus reliability, now become design parameters that are the responsibility of the design engineer.

NASA and the aerospace industry demand the design of cost-effective vehicles and associated propulsion systems. In turn, cost-effective propulsion systems demand robust vehicles to minimize failures and maintenance. Thus, the emphasis early on in this program should be effective reliability modeling supported by the collection and use of applicable data from a comparable existing system. Such a model could support the necessary trades and design decisions toward a cost-effective propulsion system development program. These analyses would also augment the more traditional performance analyses in order to support a concurrent engineering design environment.

In this view, functional area analyses are conducted in many areas, including reliability, operations, manufacturing, cost, and performance, as presented in figure 1. The design engineer is responsible for incorporating the input from these areas into the design where appropriate. The designer also has the responsibility to conduct within and between discipline design trades with support from the discipline experts. Design decisions without adequate information from one or more of these areas result in an incomplete decision with potential serious consequences for the hardware. Design support activities in each functional area are the same. Models are developed and data are collected to support the model analysis. These models and data are at an appropriate level of detail to match the objectives of the analysis. Metrics are used in order to quantify the output. Comparisons are made to the requirements and further definition provided back to the designer. This is an iterative approach that supports the design schedule with results updated from increasingly more detailed design information.

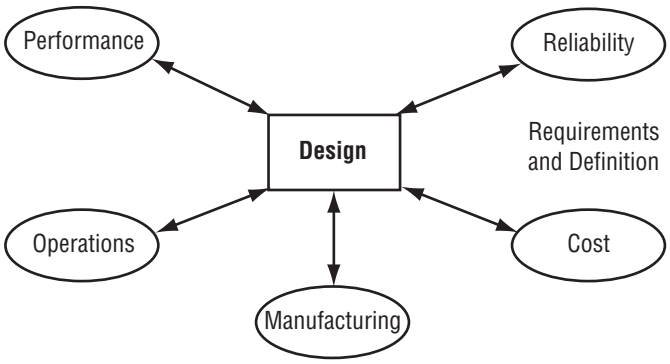


Figure 1. Disciplines in design.

Currently in aerospace applications, there is a mismatch between the complexity of models (as supported by the data) within the various disciplines. For example, while good engine performance models with accurate metrics exist, the use of absolute metrics of reliability for rocket engine systems analysis is rarely supported. This is a result of the lack of good test data, lack of comparable aerospace systems, and a lack of comparative industrial systems relative to aerospace mechanical systems. Also, metrics are less credible for systems reliability. There is, as yet, not a comparable reliability metric that would allow one to measure and track reliability as the engine Isp metric allows one to measure and track engine performance. Performance models, such as an engine power balance model or a vehicle trajectory model, tend to be of good detail, with a good pedigree, and the results well accepted by the aerospace community. The propulsion system designer has to be aware of these analysis fidelity disparities when it becomes necessary to base a design decision on an analysis. It is the responsibility of the reliability engineer to develop good reliability models with appropriate tools and metrics to rectify this situation.

There is a need to develop reliability models to obtain different objectives. Early in a launch vehicle development program, a top-level analysis serves the purpose of defining the problem and securing top-level metrics as to the feasibility and goals of the program. This "quick-look" model effort serves a purpose—it often defines the goals of the program in terms of performance, cost, and operability. It also is explicit about the need to do things differently in terms of achieving more stringent goals. A detailed bottom-up analysis is more appropriate to respond to the allocation, based on an indepth study of the concepts. The "quick-look" model is appropriate if the project manager is the customer; the detailed analysis is directed more at the design engineer. Both are of value. The "quick-look" model also may serve a purpose as the allocated requirements model, the model to which comparisons are made to determine maturity of the design. It is inappropriate to use the data that supported the allocation of requirements to also support the detailed analysis. Although often done, this is akin to a teacher handing out a test with the answers included.

2. BACKGROUND

Historically, design reliability processes and reliability validation procedures were inadequate. For example, there was interest in quantitative risk assessment for the Apollo program but the effort in this area was abandoned early on.[10] Thus, for at least 40 years, the design, development, and operation of liquid rocket engines has been based on various specification limits, safety factors (SF's), proof tests, acceptance tests, qualification demonstrations, and the test/fail/fix approach. There has never been a real hardware reliability requirement. Past system reliability demonstration requirements on the H–l, J–2, and F–I engine programs (99-percent reliability at 50-percent confidence) were not sufficient for demonstrating the reliability of such systems. A 99-percent reliability on a single engine is too low to guarantee an adequate engine cluster reliability (assuming independence, 95 percent for five engines). Although a 50-percent confidence does specify a low number of tests (69), it does not ensure sufficient confidence in the system.

The traditional aerospace vehicle design process can be characterized in four steps: (1) Design conservatively, (2) test extensively, (3) determine cause of problems and fix, and (4) try to mitigate remaining risk.

In today's environment, this process is prohibitively expensive. An approach is needed that supports conservative and effective design, ensures reliable hardware, and is cost effective.

While there have always been reliability tasks and activities, the reliability activities were always on the fringe of the mainstream design activities. This was a consequence of the priority associated with reliability relative to cost, performance, and schedule. Reliability functions such as failure modes and effects analyses (FMEA's)[11,12] were often performed after a design phase was completed. Lessons learned were often not exchanged from one program to the next. Reliability allocations or goals were not always specified. A propulsion system reliability point estimate from a comparable historical launch vehicle is generally a metric too crude to be meaningful in evaluating alternative concept propulsion systems. Moreover, reliability test requirements for the purpose of verification of reliability requirements are so extensive as to be impractical, given time and cost considerations. All these factors tend to minimize the effect that reliability engineering had on the vehicle and propulsion system design. Developers of launch vehicle systems have had to rely on the existence of design margins, intrinsic design conservatism, and extensive testing in order to develop reliable hardware.

Aerospace launch vehicle reliability engineering requires an understanding of how systems and components can fail and how such failures can propagate and/or be mitigated. A thorough understanding of failure modes and their effects and how they should be characterized is key to demonstrating propulsion system reliability. Different methods exist for analyzing single component or piece-part failures and system failures. Methods can be used to analyze the possibility of a generally benign failure propagating to a catastrophic failure. A probabilistic design analysis approach is key to understanding the nature of the failure possibility of the system. Coupled, these can be effective in providing a quantitative assessment of the system's reliability. While the use of such probabilistic analysis techniques can also reduce test requirements, they do not replace the importance of testing to demonstrate propulsion systems' reliability.

3. ISSUES

Much of the difficulty in generating meaningful reliability inputs to designers through the system engineering process comes from the lack of applicable and sufficient data. This problem, in aerospace mechanical reliability at least, is so acute that the reliability discipline is seen as more art than science, where groups of analysts labor long hours to produce "lots of 9's." It is a worthwhile objective to provide a reliability assessment using quantifiable metrics for a mechanical system. Other models, notably in performance analysis, generate good validated metrics of performance. If reliability analysis can provide the same thing, then the design inputs from the two disciplines are of equal fidelity, thus ensuring that reliability analysis is taken seriously. However, there are several issues that the reliability engineer must face in this quest to be taken seriously.

Although design efforts in many industries are faced with a shortage of directly applicable reliability data, reliability engineering methods are fairly well established for industries with high production rates, such as the aircraft and automotive industries, since ample quantities of good comparative data exist to support such analyses. The shortage of data for aerospace vehicle development efforts is more acute and an aerospace launch vehicle program faces the added complexity of trying to establish good reliability analysis methods, models, and tools with inadequate reliability databases. This serious problem places an added burden on the reliability engineer to support the design engineer in an effective design process. Key and somewhat unique issues facing the aerospace launch vehicle reliability design engineer include:

- How to make the most out of the little data available, including historical launch vehicle data and lessons learned from previous programs.

- How to use the results of relatively few tests that are of different duration and have different objectives (e.g., validate predicted performance) and different system configurations.

- How to verify reliability early in the program with only model data available. The lack of data leads to a lack of validated models.

- Under current methods, good estimates of reliability would require adequate failure information. Conversely, a good design would seek to minimize such failure information. If a vehicle is robust due to a good design, little reliability-type information will be available (with current metrics, failure data are needed).

Through the course of this TP, these issues will be discussed and suggested approaches derived, where possible. For example, the verification issue is brought up in section 4.2 with an extensive discussion in appendix A. Nevertheless, the lack of reliability data in aerospace is acute and severely limits the analysis options.

There are several reasons behind the lack of good aerospace reliability information. Most rockets are expendable; reusables are few in number; flight rates are very low; and in most cases, flight vehicles are

one of a kind, not necessarily production vehicles. Each shuttle, for example, is substantially unique in terms of parts and subsystems. Even with the shuttles, which have been flying since 1981, there are problems with obtaining good data. Section 6.4 discusses in detail the problems associated with the use of Space Transportation System (STS) quality data.

Development usually occurred with weak, if any, reliability requirements. Rocket engines are generally on the boundaries of combustion and materials technologies. Margins to trade for reliability are virtually nonexistent. Testing is not done to failure since cost is too great. Finally, commercial launch vehicle data are often not available to the public. It is often seen as proprietary information to the company. Even some ground operations data on the STS that were not explicitly requested in a contract, while being collected and maintained by a contractor, are not generally available to the Government. These are some of the reasons why good reliability data are difficult to obtain for aerospace launch vehicles and propulsion systems.

The case is often made that aerospace propulsion systems should be comparable to aircraft propulsion systems. Though nice in theory and exciting in terms of the data that are made available, this rarely holds up under scrutiny. Table 1 presents one such comparison of the two systems.

Table 1. Aircraft to launch vehicle comparison.*

Characteristics	Aircraft	STS (Orbiter)	ELV's
Structures:			
Factors of Safety	1.5	1.4	1.25
GLOW (Klb)	618	4,426	1,888
Design Life (Missions)	8,560	100	1
Propulsion:			
Thrust (Vac, Klb)	30 to 60	470	200 to 17,500
Thrust/Weight Ratio	4.5	74	60 to 140
Operating Temp (°F)	2,550	6,000	500 to 5,000
Operating Press (psi)	140	2,970	500 to 1,200
Cruise Power Level	25%	109%	100%
Mechanical:			
Specific Horsepower	2	108	3 to 18
rpm	13,450	35,014	5,000 to 34,000

* Taken from "Operational Design Factors for Advanced Space Transportation Vehicles," Whitehair, et al. IAF–92–0879

Aircraft data generally are more readily available and in the proper format with data collected from a reliability and maintainability point of view. While this data supports good model development, the question of applicability of results is more of an issue. This is especially true of rocket and aircraft propulsion systems, with major differences in configurations, environment, and operating philosophy (see table 1). Specifically, these differences include operating environment; operating temperatures, pressures and thrust; ability to idle, taxi, and loiter aircraft engines and vehicles; use of cryogenic fuels on rockets; large performance margins on aircraft; nonintrusive health management of aircraft propulsion systems; and, perhaps the major difference, a philosophy of use with aircraft that tolerates test and operational failures (and even loss of life).

It is important to note that an understanding of the reliability methods, models, data, and tools required to do the job only presents a partial solution to the traditional problem of reliability assessment not being effectively involved in the design process. Management methods are also critical in ensuring that reliability considerations are implemented in the design process. Techniques such as concurrent engineering, total quality management (TQM), variability reduction programs, and probabilistic methods must be used to ensure safe and reliable hardware. Adoption of reliability analysis methods and management techniques should provide control of key reliability drivers, design variability, and failure modes.

A final difficulty in the acceptance of reliability data into the design process is a perplexing one. A traditional reliability analysis approach has existed for some years and, while peripheral, has been somewhat accepted. This has led to difficulty in changing the system to potentially more meaningful and accurate approaches. The traditional approach relies on simple top-level models such as reliability block diagrams (RBD's)[12] and analyses such as FMEA's done by groups independent of designers.

Since traditional analyses are usually after-the-fact and used only in programmatic decision making, they are useful only from a verification perspective, not from a design iteration support point of view. Such information is generally met by the design community with skepticism and is unlikely to have an impact on design decisions. Critical to the designer is accurate reliability data available in a timely fashion each design iteration in support of design trades. Independent reliability verification and assurance is important but should not be confused with the iterative design reliability activities. Reliability assurance personnel usually have the added burden of being the customer in terms of safety requirements. This requirement inherently restricts their involvement in the product team type of environment in support of design. As should be apparent and as will be stressed throughout this TP, the reliability assessment is only as good as the knowledge of the design detail of the system. The designers are the ones with full breadth and depth of insight into the design issues.

These are the critical issues facing the design reliability engineer. They need to be satisfactorily resolved for reliability analysis, especially quantitative analysis, to play a critical and consistent role in ongoing design activities.

4. DESIGN RELIABILITY ASSESSMENT METHODOLOGY

The objective of this TP is to define the reliability modeling and analysis activities that are part of an overall strategy that will ensure the design and development of a highly reliable launch vehicle. To accomplish this, all design activities by phase are identified and placed in a top-level design flow. It must be stated upfront that this is a work in progress—the method described here is an evolution of an approach to this point. The approach taken here is proactive—reliability engineering activities are done upfront in the design process and concurrently with other design activities, such as those related to performance and cost. As stated earlier, this design reliability analysis is accomplished by analyzing the design in its "failure space." Taking this perspective allows the designers to analyze the design to focus on failure scenarios. Practical design criteria will be specified and models will be developed that will assist in verifying reliability early in the program. Component and system-level reliability models, which use all existing data as effectively as possible, will be developed. This modeling is of critical importance, since traditionally, good models have been lacking in systems reliability analysis. Indeed, the focus of this document is good mechanical reliability model development with the use of quantifiable metrics and an effective tool. As stated earlier, the goal of better, more easily measured, and quantified reliability metrics is a worthwhile one. It has been said that if something cannot be measured, it is unlikely that anything will ever be done about it.

To meet this objective, this TP:

- Lays out design phase activities.
- Lists overall activities, including reliability activities for each phase.
- Describes all activities at a top level and the design reliability activities at a lower level (descriptions deferred to app. B).
- Provides more detailed discussion, including exploration of concepts and lessons learned, for key reliability activities such as modeling and analysis (app. A).

4.1 Approach

Fundamental to methodology is an integration of the reliability activities, including modeling, into the design activities.[13–16] Reliability engineering must be conducted by the design engineers as an integral part of the design process. Along with practical design criteria, good reliability tools and models will be in place to assist this process. Some education and training may be necessary to familiarize the design engineers with the design tools. Also, management support and direction will be necessary to ensure the implementation of this approach.

If the design and management techniques are effectively adopted, the hardware design should result in fewer failures and lower life-cycle costs. To realize these lower costs, early investments are

necessary to ensure that reliability plays an equally important role with cost, schedule, production, and performance considerations. Good life-cycle cost models must accurately reflect the costs of unreliability/failure, repair, downtime, and manpower costs. This will support the importance of reliability inputs to the design process. There is clearly a direct link between reliability and operations, maintenance, and cost.

The concepts developed in this TP are directed at propulsion system structural design and development (primarily liquid propulsion systems). This is intended to include mechanical systems, such as main propulsion systems and engines, but not electronic systems, such as avionics or software systems. Applicability to these systems was generally beyond the scope of this investigation, although in section 6.4 sensor discrepancy reports are included in the analysis. However, this design approach should not be unique to propulsion systems, and its applicability to other systems should be explored in future activities. Design approaches for each and every system on a launch vehicle must be consistent and integrated from the outset.

Figure 2 provides an overview of the primary activities for propulsion systems and vehicle design and development through the operations phase. It emphasizes that reliability activities are important to each stage of design and development and should be at least equal in importance to cost, schedule, and performance. Figure 2 also provides an overview of the design reliability activities. Required reliability activities, such as prediction, modeling, and verification are identified in each appropriate phase. Definition of each activity and its scope is deferred until appendix B. It is important to note the difference in reliability allocation and reliability prediction. Allocation is a top-down partitioning of reliability to subsystems and components, based primarily on historical numbers, while prediction is a bottom-up analysis of detailed design, test, and other analytical data. Too often these reliability analysis activities have depended upon the same data. Logically, this is similar to giving a test to students with the answers on the test. It is imperative to achieve credibility—that these be independent activities.

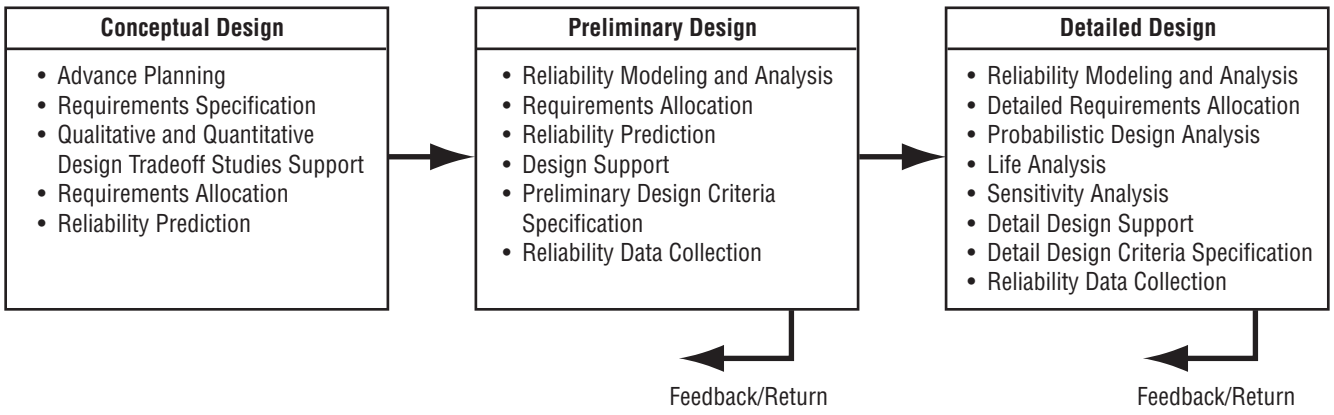


Figure 2. Design reliability activities.

Figure 3 provides an overview of the design reliability modeling approach. Key models are developed consistent with the level of detail required at each design phase in support of design estimation, trades, and sensitivities. The modeling must support the analysis-intensive activity referred to as probabilistic design analysis (PDA) which analyzes the physics of failure at the lowest level. Databases and engineering judgment are critical at each step, as are concurrent design analyses from other disciplines, including cost, manufacturing, performance, and operations. If the design is acceptably optimized between and among disciplines, the design is mature. If not, the next iteration with new detail begins.

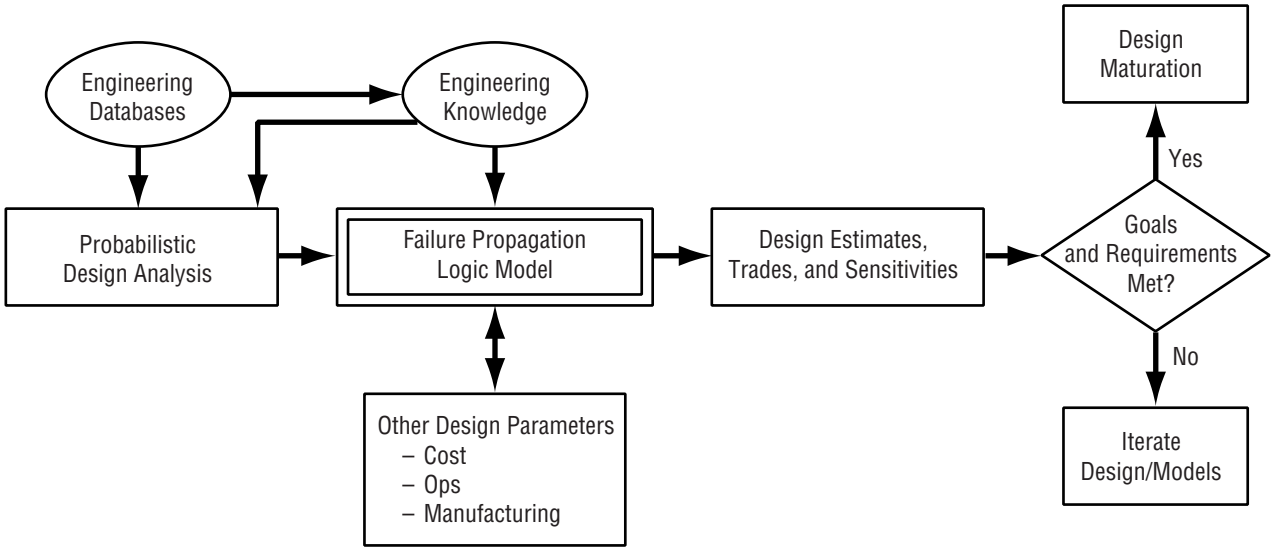


Figure 3. Propulsion systems reliability modeling approach.

The design reliability model developed to support this process (referred to here as failure propagation logic) should be a type of model that is useful in later phases of design, as this one is, and thus, may be updated within the same tool that began the process. Switching tools and models in midstream is not cost or manpower effective. Models will also need to be developed by state within each phase. Key reliability concerns will exist in flight, preflight, and postflight. Again, the same set of tools and models should be readily applicable to modeling within these separate states.

It is imperative that the process and data that the reliability engineer uses to provide reliability inputs to the designer be visible and open (as so often is not the case). Sources and quality of the data must be explicitly discussed. Any weaknesses in the data must be acknowledged. Only through this will a designer have good enough information to understand the fidelity of the input and the priority to place on it in making decisions between design alternatives.

Figures 4–6 provide overviews of the design activities occurring in the conceptual, preliminary, and detailed design phases, respectively. In these figures, "mainline" activities, or those likely to be seen on a top-level program schedule, are in bold boxes. Activities that are primarily reliability activities are in shaded boxes. Activities are always iterative and correlated. For example, reliability analyses have strong impacts on maintenance and cost activities. Many arrows that could be used to show iteration and feedback have been left out for simplicity. Between each figure (phase of activity) there would be a review phase, at which point a return to the previous phase of design activity is possible.

Figures 4–6 correlate with the text provided in appendix B, which discusses each box with most of the detail reserved for the reliability activities. References are made where appropriate. Figure titles and section titles are the same, and section numbers are shown on each box in each figure. The reliability-related activities occurring outside of the design phases are only briefly discussed in this TP.

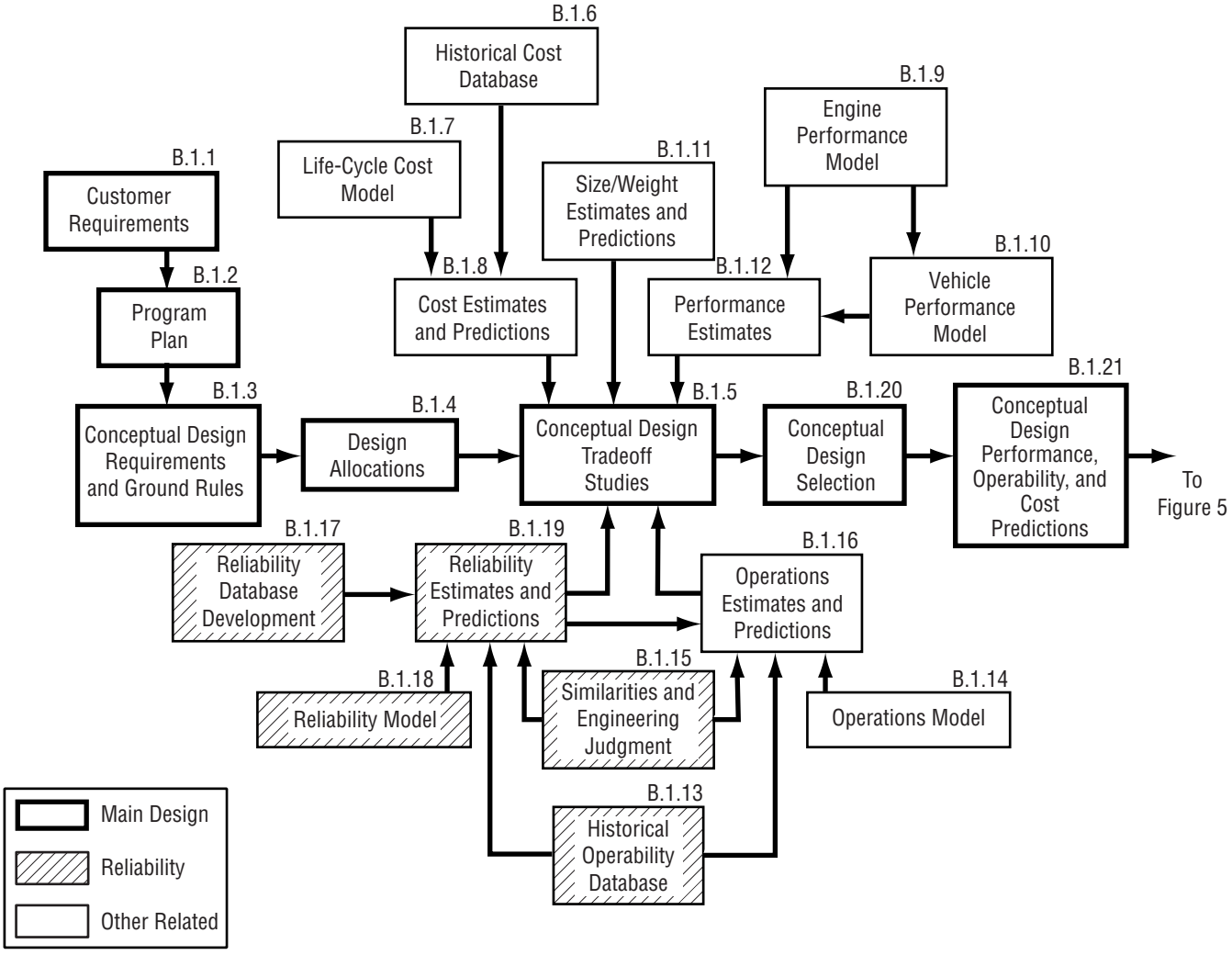


Figure 4. Conceptual design phase activities.

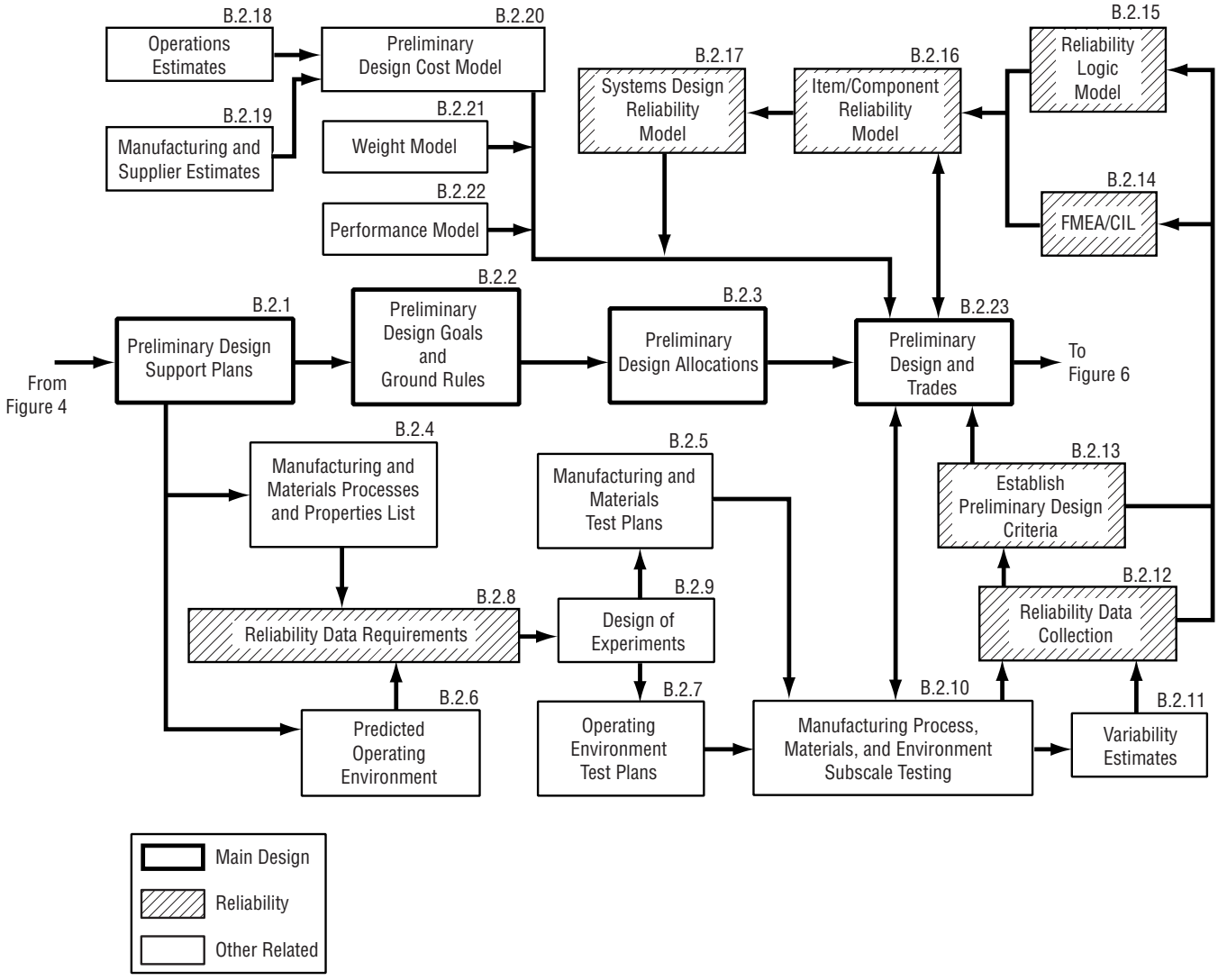


Figure 5. Preliminary design phase activities.

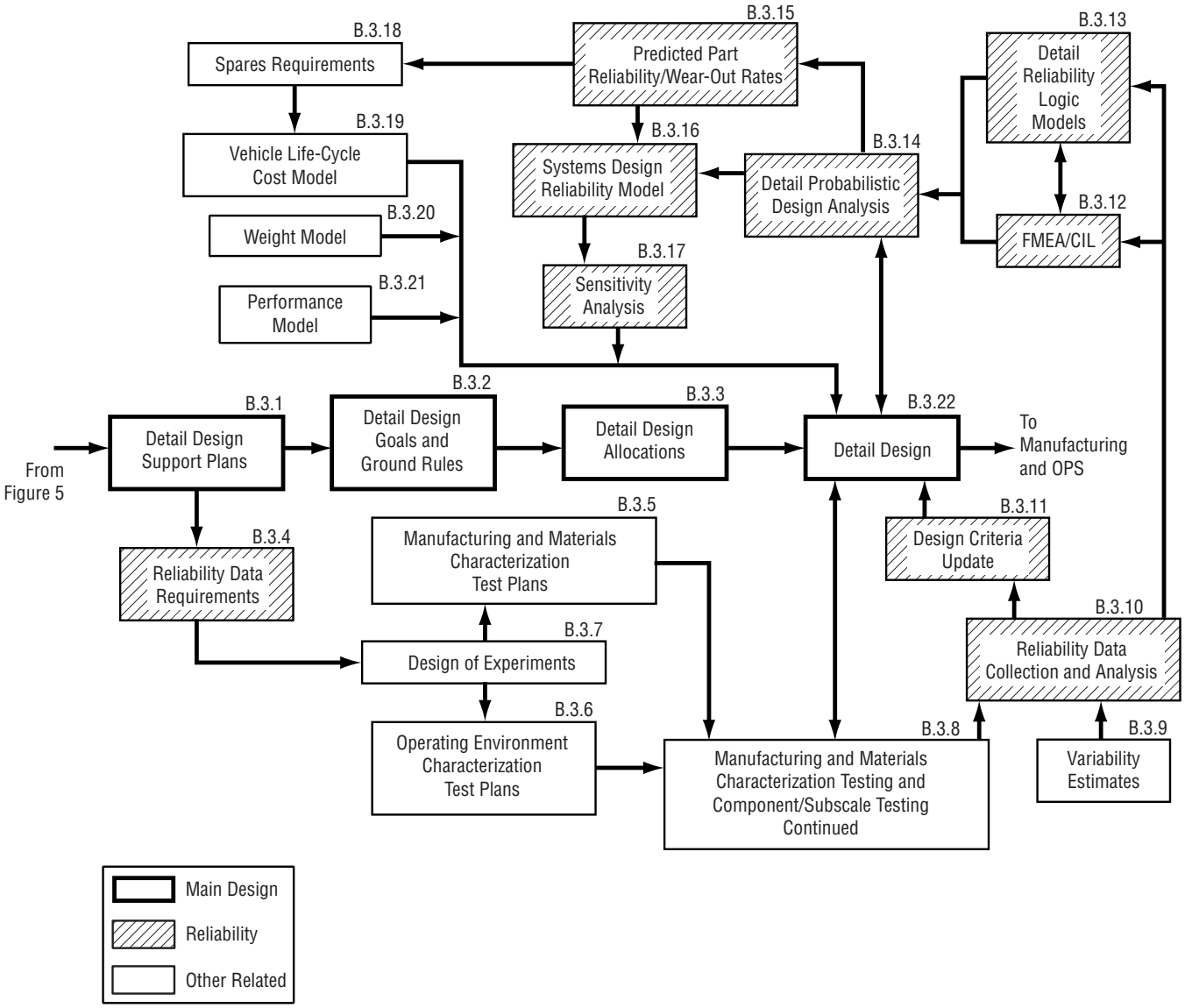


Figure 6. Detail design phase activities.

4.2 Key Topics: Design Criteria, Quality Control, and Verification

This section provides additional detailed information on selected reliability activities presented in section 4.1. Since much of the presentation is detailed with derivations involved, the analysis has been placed in appendix A. Three key areas are discussed in detail.

The ultimate goal of the design reliability engineer is to establish effective design criteria. Design criteria are considered a direct way to significantly impact the design from a reliability perspective. The goal is to establish simplified design criteria that reflect a deeper understanding of the design information (e.g., probabilistic versus deterministic) and yet, not result in major changes in the tradition methods of design, since it would be impractical to retrain and reeducate all hardware designers. The traditional design approach uses the SF as the reliability design criterion. Key disadvantages to the SF as design criterion are that it is wasteful of design resources and it does not ensure reliability. A derivation is possible of a more appropriate design criteria—the safety index (Z) that would help these problems. In this view, application of an approach using a Z design criteria will allow the design resources to be more efficiently applied to critical hardware parts and will ensure a more robust design. This approach takes a "physics of failure" view to establish design criteria that are more meaningful to the designer, more reflective of probabilistic concerns, and more related to actual reliability of the hardware. The extensive analyses and derivations on this topic are presented in section A.1 of appendix A.

Another key topic is the discussion of the quality control (QC) process and its ability to ensure reliability. A discussion of traditional aerospace QC finds serious shortcomings in this regard. Section A.2 develops a new way to look at QC and derives an important concept referred to as the QC design margin. Application of the QC design margin in an effective QC process will improve the chances of selecting reliable hardware.

Finally, how does one attempt to verify the reliability of hardware? Traditional design verification approaches have included binomial and reliability growth modeling. The focus here is on developing a new verification approach that is consistent with "physics-of-failure" modeling. The purpose of testing then becomes the verification of these models. This type of engineering model verification is more appropriate and realistic than a general statistical model verification which requires an enormous amount of test data with tests to failure. This discussion of reliability verification and the development of a "physics-of-failure" modeling verification approach are presented in section A.3.

This section just scratches the surface of a very challenging area, and much additional work needs to be done. Concepts that could be further developed and explored include: modeling approaches for extreme values, correlated failure modes, system wear-out, critical failure mode identification, and reliability growth (test-fail-fix). Other areas that could have significant impacts on system reliability and should be examined include proof and acceptance testing, reliability data system definition, and malfunction warning systems.

5. MODEL AND MODELING TOOL DEVELOPMENT

At Marshall Space Flight Center (MSFC), the need for reliability in design engineering became increasingly important in the early days of the National or New Launch System (NLS). This need has become significantly more important in the X–34, X–33, and RLV programs. At the time of the NLS, most of the available methods and tools were either inadequate for the required analyses, required the use of multiple tools for a single analysis, or were inappropriate for use by design engineers. Thus, the need for new methods and tools for conducting reliability analysis was realized and led to the initiation of enhancements to an existing software package to meet their requirements. The results of these software enhancement efforts to date represent the failure environment analysis system at MSFC (FEAS–M).

With decreasing budgets and the need for greater commercialization of launch vehicle services by the United States in recent years, launch vehicle systems reliability, and reliability analysis, has become increasingly important. During the Apollo days, systems reliability and risk assessments lost favor in the design and program management arenas due to the lack of full understanding of how to conduct a meaningful analysis.[10] This resulted in the adoption of the FMEA and critical items list (CIL) method of risk management for the STS.

After the *Challenger* incident, it became apparent that the FMEA/CIL method, as implemented, was inadequate. This method does not allow a meaningful quantification of systems reliability or risk for launch vehicle systems. As a result, a resurgence of systems analysis and probabilistic risk assessment (PRA) was realized. The proper implementation of methods and tools, such as fault trees and event trees, can be used to meet many of these systems modeling and analysis needs, but there are also limits to their capabilities. One of the major shortfalls of these methods is the requirement, for quantification purposes, that all initiators be independent events. Due to the high degree of correlation between failure modes in launch vehicle systems, a meaningful detail systems model cannot be constructed with these tools. This is especially true in liquid propulsion systems. For high-level modeling, a very skilled and knowledgeable analyst can develop workarounds for these shortfalls by properly selecting the definition of the basic events to eliminate much of the correlation and then modify the data for quantification of these basic events, such as to minimize correlation effects. This type of analysis is quite effective in assisting program management in the decision-making process at the higher levels of management, but is inadequate for the design engineer to make component and part-level decisions during the design and development stage.

Another method of quantifying reliability which has seen a resurgence in popularity in the aerospace community is PDA. This method is quite detailed and generally requires some degree of formalized training. PDA is almost always performed at the detailed part failure mode level. This type of analysis is excellent for a maturing design where detailed knowledge of the physics of failure can be gathered, but is inappropriate for conceptual and preliminary design phases. It can also be quite resource intensive depending, on the complexity of the design. An alternative method that builds upon the PDA principles but is much less resource intensive is presented in the discussion of design criteria (app. A, sec. A.1.) Usually the pure form of PDA efforts is only undertaken for high-risk failure modes. This quantification method

can only be used to feed systems models when it can be shown that the mode is not correlated, or if the correlation has been included in the PDA.

These major and multiple other minor to moderate deficiencies led to the initiation of the software development efforts for FEAS–M.

5.1 FEAS–M Design Reliability Tool

This section describes the requirements for the reliability design tool that was ultimately developed in-house, the review and selection process, and a brief discussion of FEAS–M features and performance. Reviews were held of software on the market. Because these existing products were found lacking in functionality and applicability to the typical aerospace design problem, a design reliability tool development activity was undertaken.

5.1.1 Tool Requirements

The MSFC Propulsion Laboratory embarked on a search to find and evaluate available tools for systems reliability analyses. In order to evaluate the tools, a set of requirements based on the needs of the Propulsion Lab was established.

One of the first and foremost requirements for the tool was that it must be a tool for design engineers. This meant that it must have an easy-to-use graphical user interface with point-and-click and drag-and-drop model construction without stringent model formatting requirements or extensive tabular input. Tabular input for description/development of the model was deemed unacceptable. It also required that some method or interface for relating the model to engineering drawings be included. As a design engineering tool, construction of the models and subsequent analyses had to be a fast and efficient process to avoid overburdening the engineering staff. Due to limited resources for training, the tool was required to be very intuitive and have a short learning curve. The goal for time-to-software proficiency was set at 1 wk with modeling proficiency at 1 mo. Analysis of a typical model of 1,000 events should be completed in <8 hr.

With multiple design engineers responsible for various areas of a system, the tool was required to provide some “systems engineering” capabilities, meaning that the tool must allow multiple people to work on the same model at the same time from different computers in different work areas. It also required that there be a capability for storing or linking supporting information to the model and analysis. The tool could not have a limit on the size of a model.

Due to the very high reliability of aerospace hardware, the tool was required to have accurate quantitative analysis capabilities. The tool should have a minimum of 64-bit (double) precision.

The tool should have the basic fault-tree capabilities of top-event point probability calculation, and minimal cutset generation and quantification. In addition, the tool should be capable of propagating statistical distributions of the probabilities for uncertainty analyses. The tool should allow the user to decide how common causes are treated, including treating each occurrence of the same common cause as either an independent event or treating all occurrences of the common cause as a single event. This allows the user to conduct common cause sensitivity analyses.

Launch vehicle reliability changes as a function of accumulated operation time. The tool must be capable of analyses with time-to-failure distributions and allow for analysis with the accumulation of existing service time. In addition, launch vehicle operations have multiple phases with different failure scenarios and different environmental conditions. This requires the capability of modeling and analysis of these changing conditions. Thus, the tool must incorporate the capabilities of phase-state transition modeling where subsequent states are conditioned on previous events. Due to launch vehicle reusability, the tool must be capable of accommodating reconfigurable and repairable systems.

All failure modes in a launch vehicle system cannot be considered independent events. Varying degrees of correlation exist between hardware and environments. A fundamental example of correlation is a device that controls its own loads which are continuously changing, causing changes in the strength of the device. This results in a stress/strength correlation. There are also many cases where extreme value analyses are required. An example is a group of pipes that are subjected to the exact same loading. Many correlation, extreme value, physics-of-failure, and other PDA problems may be encountered in the modeling of a system. The tool must be capable of handling these types of problems.

Due to the similarity of models for such top events as loss of mission, loss of vehicle, and loss of crew, the tool should be capable of handling multiple top events of interest within a single model. This eliminates the need for duplicating and modifying a model to achieve a similar top event and eliminates the need to maintain multiple similar models.

Due to the high number of Macintosh users in the design groups, the tool should be Macintosh operating system based. Later it will be ported to the PC environment.

5.1.2 Tool Evaluation

Ten software packages were evaluated against the requirements. The tools evaluated were:

1. Automated Reliability/Availability/Maintainability (ARAM): Computer Sciences Corportion, NASA Langley (LaRC).
2. Computer-Aided Fault-Tree Analysis (CAFTA): Science Applications International Corportion (SAIC).
3. Computer-Aided Reliability Estimation, third generation (CARE III): NASA LaRC.
4. Event Time Availability, Reliability Analysis (ETARA): NASA Lewis Research Center.
5. Failure Environment Analysis Tool (FEAT): Lockheed Engineering & Sciences, NASA Johnson Space Center.
6. FaulTrEASE: Arthur D. Little.
7. Fault Tree Compiler (FTC): NASA LaRC.
8. Numerical Evaluation of Stochastic Structures Under Stress (NESSUS): Southwest Research Institute.

9. Reliability/Availability (RELAV): Cal Tech, NASA Jet Propulsion Laboratory.

10. Semi-Markov Unreliability Range Evaluator (SURE), with Abstract Semi-Markov Specification Interface (ASSIST), Pade Approximation with Scaling and Scaled Taylor Exponential Matrix (PAWS/STEM): NASA LaRC.

None of the tools at the time of the evaluation met all the requirements. Most were basic fault-tree, direct graph (digraph) matrix analysis, RBD, or Markov analysis tools. It was apparent that the MSFC Propulsion Lab would need to develop their own tool to meet their requirements. It was decided that the FEAT software package would be used as a starting point. This software was developed under NASA contract, therefore the source code was available without cost to the Propulsion Lab. This package has excellent user interface and qualitative analysis capabilities, based on the digraph matrix analysis method. The capabilities of the existing FEAT software package at the time of acquisition were as follows:

- Point-and-click and drag-and-drop model construction with tabular input of node text block information or selectable text from tables.

- Free-form model development allowing the user to develop the model top-down, bottom-up, middle-out, side-to-side, or any other conceivable two-dimensional arrangement.

- Any drawing that can be saved as a PICT or PICT II file with entities grouped according to specific rules can be linked to the logic model.

The tool has a very short learning curve. The average beginner can begin building and analyzing models within 8 hr of their introduction to the software and become proficient with the software within 2 wk. Analysis of a 1,000-node model to find all single- and duel-point failures can be completed in <5 min on a typical desktop computer.

The software allows many users to develop portions of models that can all be linked into a single model if certain development rules are followed. This is accomplished through the use of individual model files representing a portion of the overall model. This primarily involves following a common node and file-naming convention that can be administered through the software text tables. The software allows users to link up to 10 “databases” to each “component” as defined in the PICT file. The size of the models is unlimited by the software, but may be limited by the amount of computer memory available.

The FEAT software package can graphically show the propagation of source analyses (select a node on the graphic model and propagate its effects through the model/system) and target analyses (select a node on the graphic model and determine what nodes in the model/system can cause it to fail). Effects of specific failures can be determined by setting a node to a failed state, then reconducting source and target analyses. Paths between nodes and duel-point failure partners can be shown, in addition to target node intersections.

A text file of the reachability information can be output for use in the development of an FMEA. Multiple top events can be developed and analyzed within the same model. Analyses can be conducted on any node in the model or on any “component” failure in the PICT file.

The shortcoming of the FEAT software is that it has no quantitative analysis capability.

Construction of logic models is a drag-and-drop, draw a line, and select the text process. Node structures are represented in a tool bar for quick construction access. Edges (the connections between nodes) are drawn by a simple point-and-click process. There are few set rules on how a model looks. Thus, the model can be drawn to represent a fault tree, classical digraph, or any form the user chooses. The use of a digraph representation eliminates confusion between "AND" and "OR" gates. Text blocks for the nodes are generated by a point-and-click selection method from predefined tables or by appending the tables. This eliminates typographical errors in the models.

Figure 7 represents a simplified digraph in FEAS–M, implemented as an example failure propagation logic model. Basically, a failure propagation logic model shows the flow from the lowest level (leaf node failure mode) through any intermediate stages (e.g., redline or redundancy mitigation) to a final top event of interest (e.g., catastrophic failure). This is unlike an FMEA in that typically an FMEA does not include any intermediate stages and, thus, is usually seen as a "worst-case scenario." From figure 7, either a failure in a turbopump bearing or the bearing cage leads to an intermediate pump failure that, coupled with a safety system failure in an "AND" gate (if both occur), leads to a catastrophic failure of the pump. This is a straightforward boolean logic implementation. Information critical to the development of such models is extensive and includes the following:

- System configuration data.
- Engineering expertise.
- Description of health management functions.
- Vehicle interface conditions.
- Applicable FMEA/CIL's, hazard lists, failure information.
- Failure reports of similar systems.
- Existing failure propagation logic models of similar systems.

Extensive model development from scratch can be time consuming and labor intensive. It is critical to be able to draw from a library of previously developed failure propagation logic models of key components. Such a library for propulsion systems is under constant development.

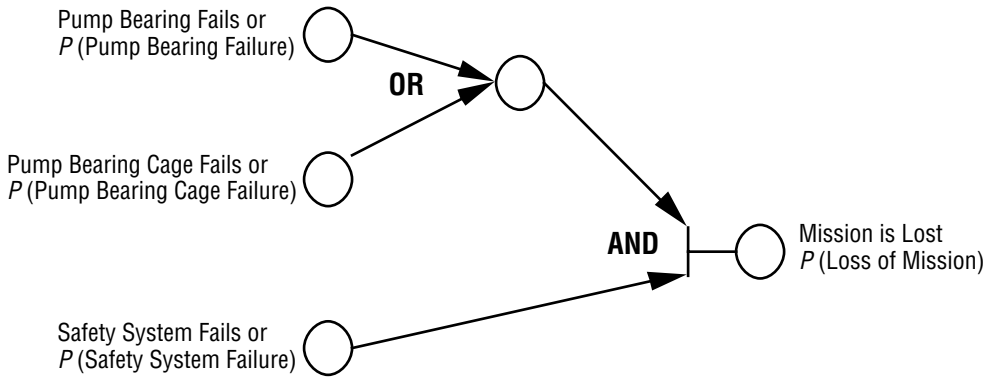


Figure 7. Model representation.

The failure logic model can be linked with a system drawing. Failure propagation is highlighted by color changes on both the logic model and the drawing. The drawing can be any file that can be saved in a PICT format, but must adhere to a specific grouping and naming convention. Linkages between the logic model and the drawing are achieved by a consistent naming convention. Analyses can be conducted either from the logic model or the drawing.

Figure 8 is an engine schematic with key components such as valves, preburners, and turbopumps, labeled so that they can be linked directly to the failure propagation logic models implemented in FEAS–M. Through the use of color, links and changes in either one are reflected in the other. Such a dynamic analysis capability results in excellent presentation and traceability characteristics for a design analysis.

The FEAS–M software allows multiple top events to be developed in a single model without using a dummy node top event. This minimizes the amount of model duplication and revision when modeling many similar top events. Nodes can branch outward to represent a common cause, minimizing or eliminating the need to duplicate the common cause node at each occurrence within the model.

A model can be constructed from many individual files or submodels. Many engineers/analysts can work on the same model simultaneously by working within the files for which they are responsible. These individual files are automatically linked back to the master model. Links within models can exist in many files at all levels of the model, not just at the top and leaf nodes for each file.

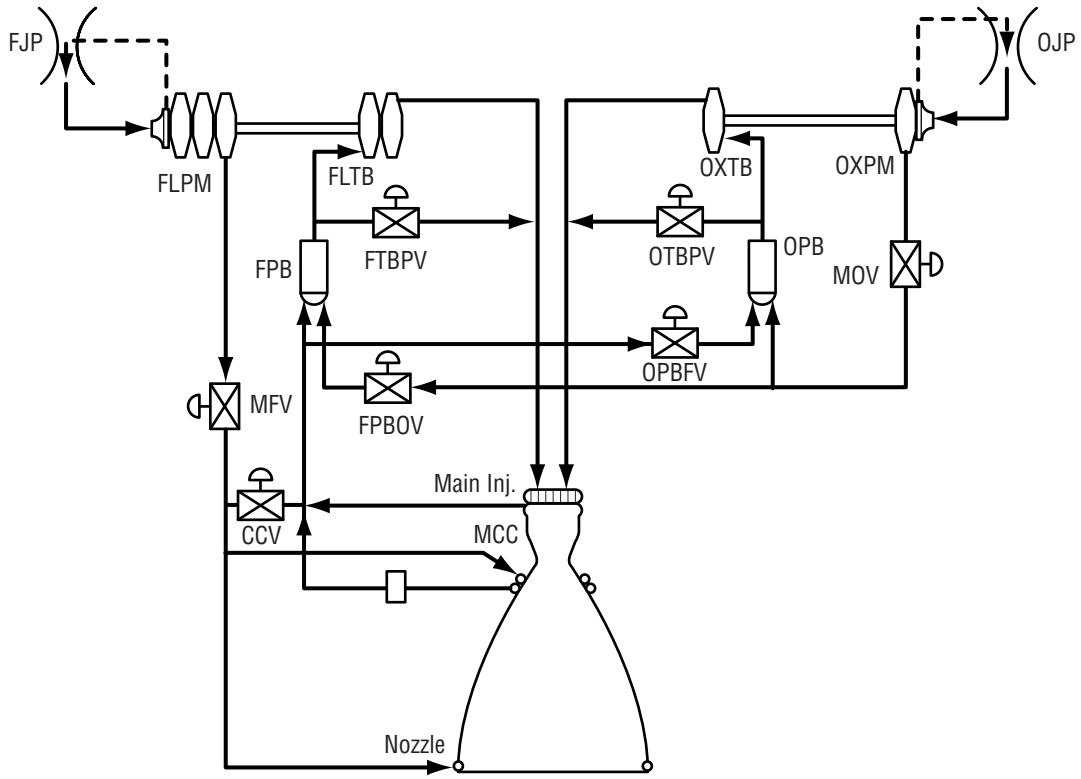


Figure 8. Model engine cycle schematic.

The model can link to as many as 10 "databases" through the drawing. Any information that can be stored as an ASCII text or PICT file can be linked to a "component" in the drawing by following a simple file-naming convention. This allows the modeler to store supporting information for the analysis within the model. This significantly reduces or eliminates the need to maintain separate databases of information. It also allows for quick and easy access to references.

The use of extensive graphics for representing the model and analyses makes this software an excellent tool for communication between engineers, and between engineers and management. The fast graphics and extremely fast computations allow for real-time "what-if" analyses in presentations and communication meetings using models with thousands of nodes.

The software identifies single- and, dual-point failures, minimal cutsets by two methods, paths between nodes, intersections of paths, and dual-point failure partners within the model and the drawing by color highlighting. Likewise, source and target analyses can be depicted. Nodes can be "set" to a failed state and their effects evaluated. This allows for evaluation and visualization of system degradation for fault tolerance, common cause sensitivity, and other what-if analyses.

The software will output the basic information required for an FMEA if the model is so constructed. This output is in ASCII text format for easy importing into the modeler's FMEA database/software or most all word processors for formatting to the requirements of the company, program, or project.

5.1.3 Enhanced Software

The expansion of the extensive qualitative analysis capabilities of the FEAT software, discussed in detail in the previous section, including extensive quantitative analysis capabilities, has been conducted. This has led to the creation of the FEAS–M software, resulting in a tool that is state-of-the-art, in the author's opinion, and supports extensive qualitative and quantitative design reliability analyses.

The point probability and minimal cutsets of any nonleaf node in an FEAS–M model can be calculated. Capabilities to expand the functionality and facilitate quantitative analysis include the following:

- Top event probability.
- Cutset generation and quantification.
- Time domain analysis.
- Probabilistic design analysis.
- Correlated failures.

The FEAS–M software has been used and is currently being used by multiple NASA Centers and contractors on programs such as the NLS, Space Shuttle main engine (SSME), RLV, and X–33. The following is a brief description of some of the FEAS–M capabilities.

FEAS–M computes the probability, cutsets, and cutset probabilities for any nonleaf node the user selects. This can be accomplished by use of the failure logic model or the drawing. Cutsets and probabilities can be calculated treating common cause nodes as individual independent events or as single common causes, allowing for common cause sensitivity analyses.

In addition to point probability propagation, the software will also propagate time-to-failure distributions and frequency distributions. Normal, lognormal, uniform, exponential, two-parameter Weibull, and three-parameter Weibull distributions are supported. Current plans also include the four-parameter Beta distribution, but this has yet to be implemented. Time-to-failure distributions are propagated by sampling the leaf nodes for a user-defined number of time intervals over a user-defined "mission duration." The modeler can also add existing service time to the leaf nodes to evaluate part replacements and mixing of parts with various use time.

Figure 9 provides an example of a time domain analysis conducted in FEAS–M. In this example, time-to-failure distributions are selected for the pump bearing, cage, and safety systems. Selecting an analysis start time (user-defined service time) and implementing the analysis (stepping through in time a user-defined number of steps) generates the top-level distribution of time-to-failure for the catastrophic failure of the pump. The impacts in changes in time-to-failure distributions (perhaps reflecting maintenance) at the lowest levels can be immediately seen in the top-level event of interest.

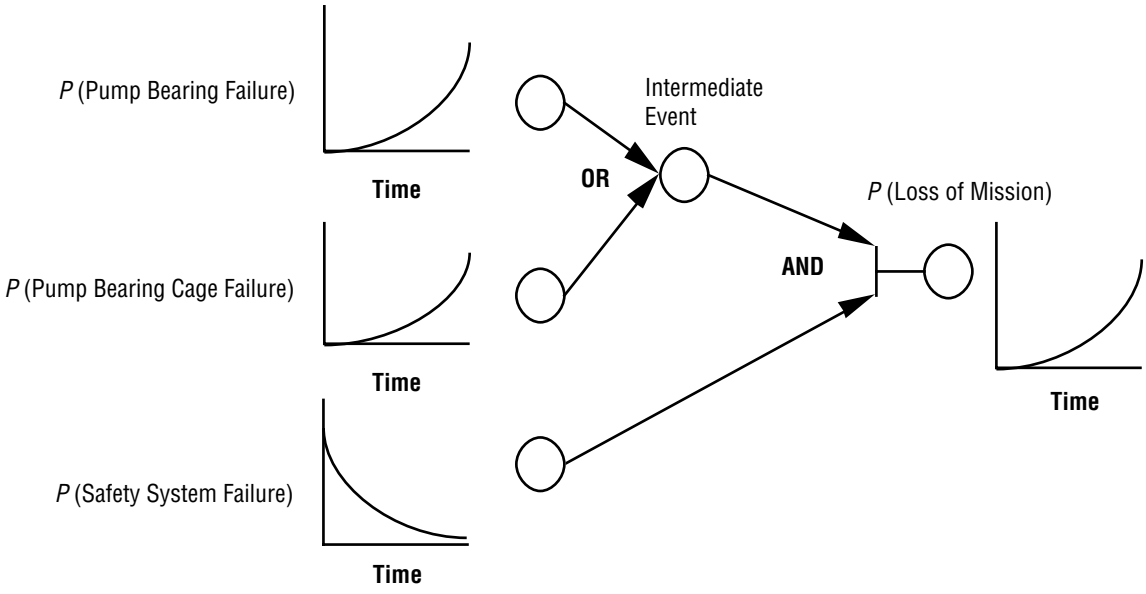


Figure 9. Model time domain analysis.

FEAS–M also incorporates the basic capabilities of PDA, accomplished through the use of user-definable equations or equation gates. These gates combine the values of the input nodes using the algebraic operators for addition, subtraction, multiplication, division, and exponentiation. For PDA, FEAS–M performs a Monte Carlo simulation on the leaf nodes, propagating the values through the model to the selected top event. The equation gates can also contain logical operations. "IF-THEN-ELSE," "AND," "OR," <, >, and = are supported.

Figure 10 provides an example of a PDA implemented in FEAS–M. In this example, through careful PDA modeling and analysis done off–1ine, it was determined that a particular turbopump part's (liquid oxygen (lox) damper seal) stiffness is determined by three key attributes: seal exit clearance, seal inlet clearance, and the change in pressure across the seal. The relationship can be explicitly specified and is

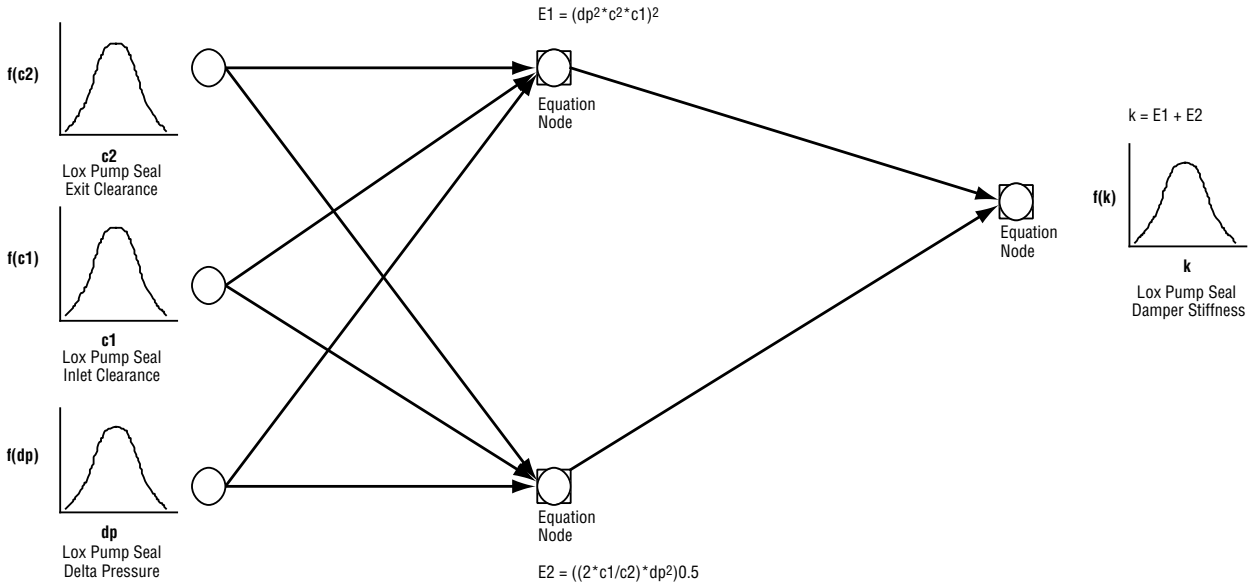


Figure 10. Model probabilistic design analysis support.

represented in the equation nodes. The relationship is split into two equation nodes only for presentation purposes. The top-level event of interest (pump seal damper stiffness) is generated from the distributions of the lower attributes in a Monte Carlo environment provided by FEAS–M. Thus, FEAS–M can support the tool attributes needed to support PDA input to the failure propagation model analysis.

The use of equation gates in the FEAS–M software allows the user to define correlated failures. Boolean logic modeling tools require that all leaf nodes be independent events. This software allows the user to define the correlation using the equation gates, thus eliminating the need to group or dissect failure modes to create independent events.

The "set" and "source" functions mentioned in the Qualitative Capabilities section above are also supported in all the quantitative analyses. The common cause treatment selection is also applicable to the time-to-failure distribution analyses.

Figure 11 provides a summary of the quantification capabilities built into the FEAS–M model. Included, as discussed, are the PDA support (equation gate), time domain analysis, failure probability point estimates, cutset designation and probabilities, and a list of distributions currently available and supported.

FEAS–M is very memory efficient: the application requires <1 Mb of hard-drive space. A typical 1,000-node model requires <3 Mb of hard-drive space. Analysis of a typical 1,000-node model requires <4 Mb of RAM. The recursive algorithms used in the program are very fast and efficient. These algorithms do not implement any approximations. FEAS–M can solve the probability, cutsets, and cutset probabilities for a typical 1,000 node model in ≈1 sec on a typical desktop or notebook PowerPC.

The software and user's manual exist and are available upon request.[17] Sections 7.1 and 7.2 present specific examples of qualitative and quantitative design reliability analyses that use the FEAS–M software.

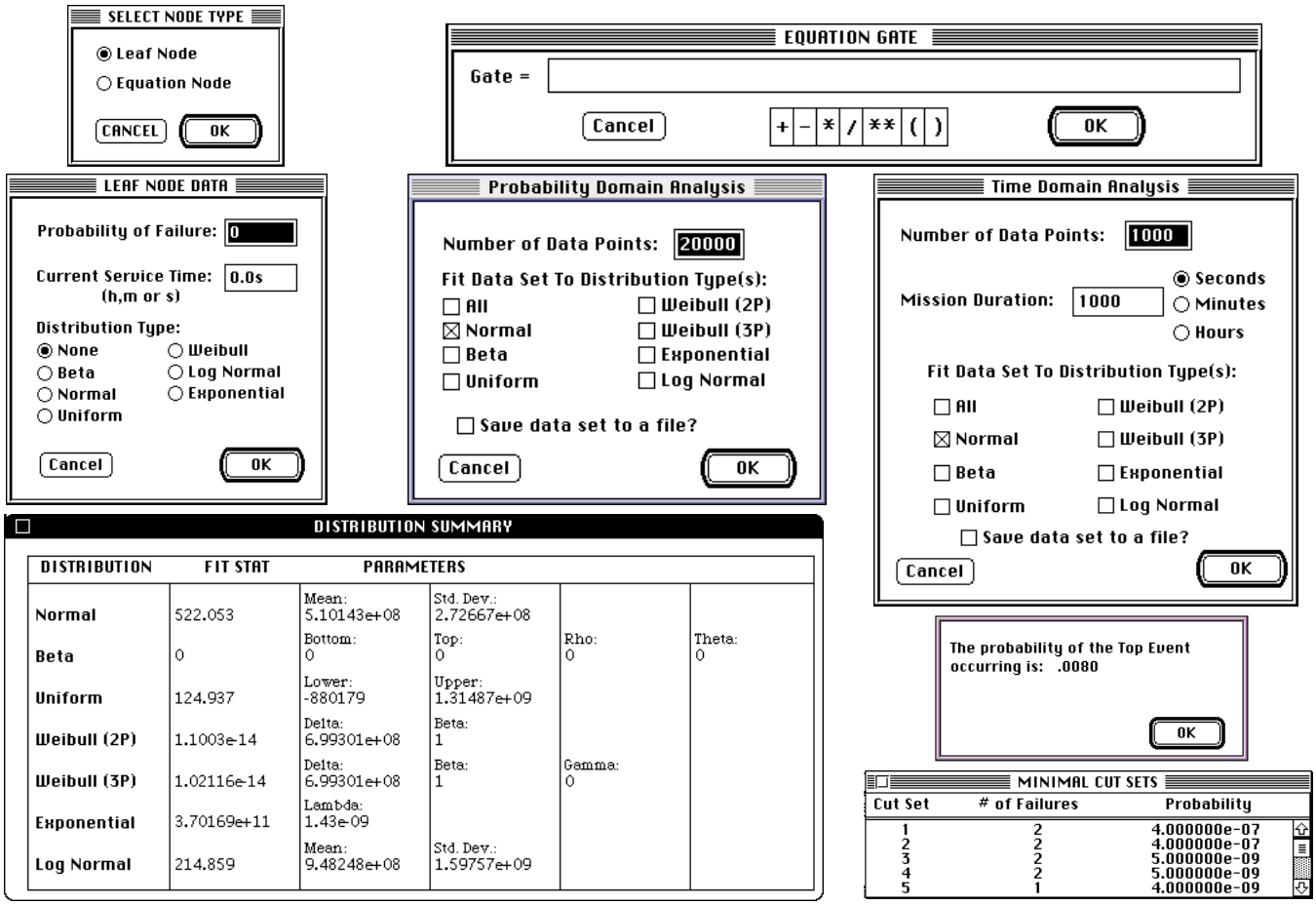


Figure 11. Current quantification capabilities.

6. BASIC ISSUES IN QUANTIFICATION

Section 3 discussed general issues related to design reliability, including lack of data, comparisons against aircraft, and the suitability of current approaches, among others. This section expands upon this discussion to describe frequently used and available data sources, applicability of data issues, and takes an indepth look at the most frequently used database from the Shuttle. The ultimate goal in model quantification is to accurately provide information on the probability of failure for reliability predictions. Ideally, this includes not only point estimates but confidence intervals as well.

The type of quantification in discussion is that coming from systems modeling as opposed to a deterministic or probabilistic design analysis of a piece part, a component, or a particular structural material. Excellent discussions of the latter approaches appear in several sources.[18–21]

The use of reliability predictions for space hardware has grown considerably in the past 10 years Several sources present predictions or assessments on STS hardware.[22–26] Even though the authors stress that the results are more qualitative than quantitative in nature, the results are often perceived as absolutes.[27] This kind of analysis is often referred to as probabilistic risk assessment. The need and popularity of this activity is growing and models and tools to support this are in high demand.

In the propulsion systems world, the identification and assessment of applicability of data is a very difficult process. Indeed there is some controversy as to whether the quantification of systems reliability for aerospace propulsion systems should even be undertaken and to what degree.[28] There are, of course, other approaches to quantification.[5, 29, 30] These often utilize simpler, more straightforward failure databases and systems models (e.g., growth and Markov models) to generate failure rate estimates, but do not generate the detailed hardware failure rates that are so in demand. Unlike the electronic world, where good databases, models, and methods exist,[31] for the PRA type of quantification, there is a dearth of good propulsion mechanical reliability databases. On the electronic side, many inexpensive and applicable tests can be run to gather such data. Propulsion systems testing requires, in general, extensive facility support, complex test articles, expensive propellants, and considerable manpower to set up. Such sporadic testing does little to chip away at the large need for applicable data. Indeed, the difference in applicable support data for the electronic and mechanical design reliability problem is so striking that the disciplines, in the authors' opinion, may be considered distinct.

For parts and components that are more common and more likely to be used in a commercial environment, obtaining applicable data is more straightforward. For example, relatively good data exists for feedlines, valves, ducts, and actuators. Such hardware is in use commercially, although perhaps not in a similar space environment. Conversely, next to no data exists on the reliability of flight-weight combustion chambers. These are few to begin with and few comparable systems in industry that support comparison. This point will be illustrated in an example in section 7.2.

Another traditional problem with aerospace reliability design analyses or estimation is that human factors (errors) are typically ignored or only implicitly included in the failure data. Since 20–80 percent of

errors in complex systems[32] can be due to human or process error, an omission of human factor data is a serious shortcoming. No doubt, some reasons for its exclusion include the difficulty in modeling human factors, shortage of data, emphasis on hardware only, and difficulty in carrying reliability impacts across phases. For example, given an error in manufacturing occurs, how will this affect the flight reliability of the specified hardware? This is an area in aerospace that needs a lot of effort. Markov models are often used to model phase relationships but are infrequently used in aerospace applications. The coupling of Markov models with PRA type of hierarchical fault propagation models would require an enormous effort with the benefit unclear. Again, lack of data plays a role.

Thus, it is easy to understand the attractiveness and widespread use of the unsatisfactory condition reports (UCR's) collected and used within NASA.[33] Such data are collected each test, flight, pretest, preflight, posttest, and postflight and used in the calculation of failure rate. The UCR database provides an extensive record of "problems" associated with propulsion system hardware. Over the course of the STS program, an extensive database has been built. Based on size alone, this database appears to support statistical calculation of failure rates and confidence intervals. The numbers are often used as probabilistic inputs or as weights in calculating hardware failure rates. However, as will be seen in section 6.4, the quality of the data is in question for this purpose.

6.1 Quantification Methodology

Figure 12 presents an overview of the quantification data methodology. Data are collected, if possible, by failure mode. This is necessary because different failure modes can lead to different outcomes. For example, a valve (e.g., a prevalve) failing open during flight might have insignificant consequences; yet if the same valve failed open during ground operations, there could be serious risk to operations personnel. Also, if a prevalve fails closed inappropriately during flight, the flight would be terminated. A discussion of what failure modes are important during what phases of operation and a feel for the severity of the problem would be supported by an FMEA.

Historical databases are searched for applicable data, including NASA, Department of Defense (DoD), and commercial sources. Comparisons are made based on configuration, environment, materials, and manufacturing. If data are available, it is databased and formatted for use. If not, it is identified for collection and update.

If surrogate and/or historical databases do not appear to be applicable, other sources are searched. In the case of components, piece parts, or structures design, analysis data may be available. As discussed in section 4.2, this is the physics-of-failure type of data generated by extensive and costly stress/strength types of models and tests. Rarely available due to its resource-intensive requirements, it would be the best type of data available for part and structure reliability. It is probably available only for parts deemed very critical and at high risk to the success of the system.

As a system undergoes development, data are being collected through the testing phases. The data being collected are driven by test data requirements which will also have input into the test plans and the number and types of tests. These data feed the calculation of the failure rate estimates, either in conjunction with the historical data and/or the design analysis data. Rate adjustments are, in general, discouraged here, but may be appropriate to stress the qualitative nature of the estimates.

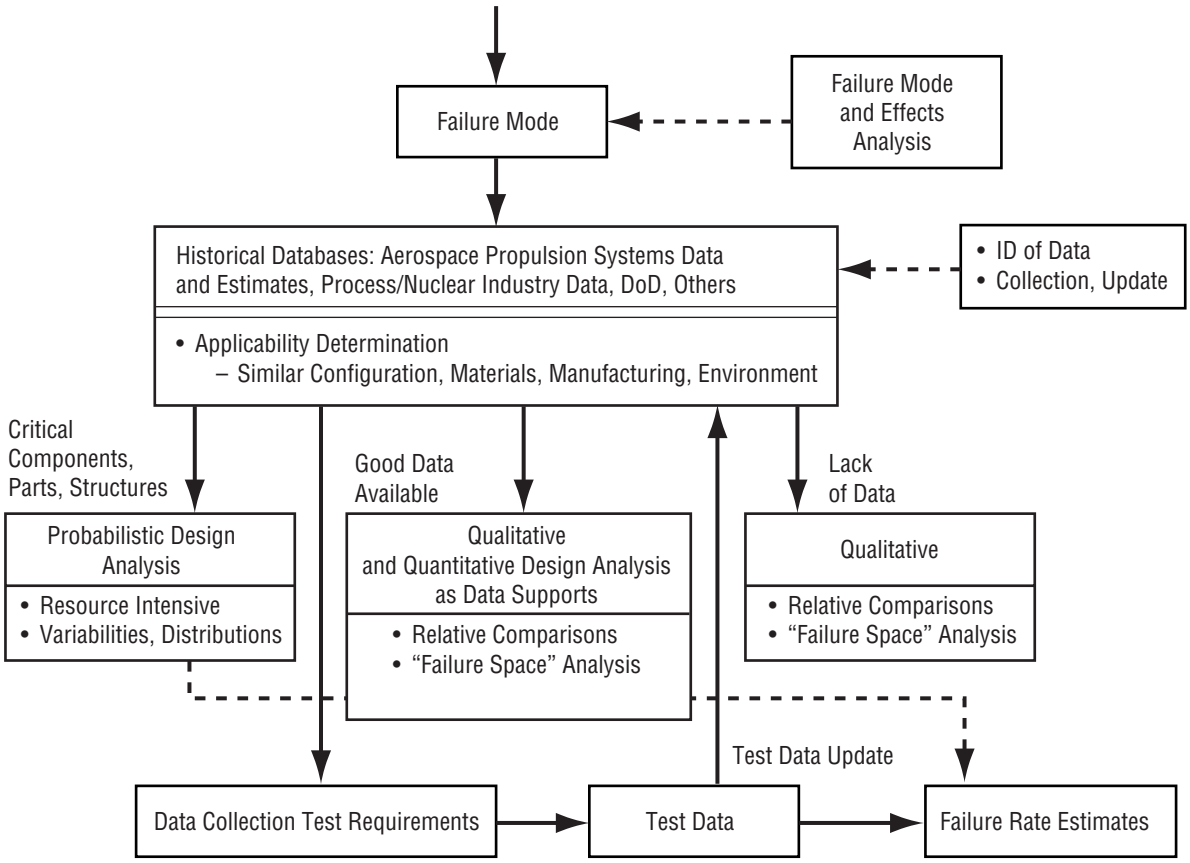


Figure 12. Quantification data and analysis methodology.

Finally, the use of human factor data in design reliability analysis is important. The selection of models and tools that allows reliability impacts to crossover phases (key to human factor) issues must be supported. Though this area is not typically modeled in aerospace applications, it is likely that it will have a large impact on failure rate calculations.

6.2 Sources of Data

Good reliability data are the backbone of accurate design reliability modeling. Without good data, modeling is, at best, incomplete. This section discusses the types of data available to the aerospace design reliability engineer and comments on its usefulness. Figure 13 presents the general data collection and analysis approach with model requirements, and the model specified as a means of establishing data requirements. Knowledge of the model requirements defines the level of detail required in the data collection process. It also serves to identify the data that are missing and should help to allocate resources to initiate activities for its collection.

Several sources provide a good discussion of the references available for mechanical reliability data, including aerospace information. One good data source that provides 50+ references is Dhillon,[32] pp.163–171, which lists many nonaerospace and aerospace data sources. Other specific and important

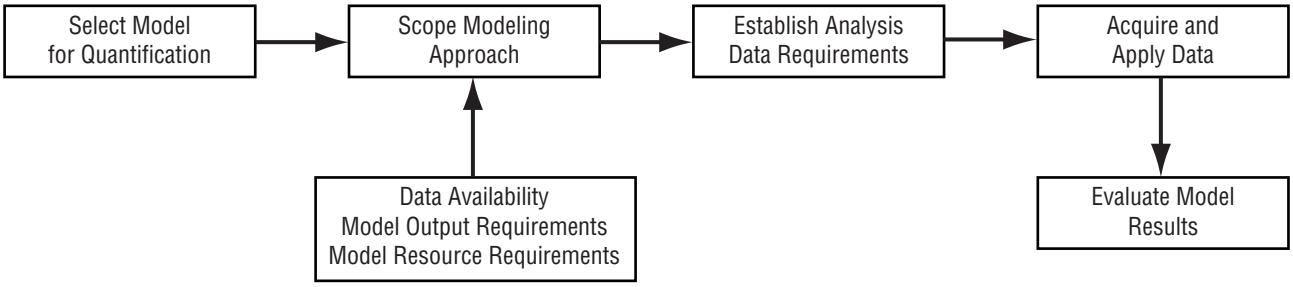


Figure 13. Model data collection and analysis.

aerospace-related data have been collected and appear in this TP's reference section.[34–39] These data provide, for the most part, the best information available relative to nonelectronic parts and systems such as valves, feedlines, bearings, pumps, and engines. A discussion of mechanical systems reliability would not be complete without considering human factors as well. Since a significant percentage of the problems appearing in mechanical systems that require human intervention are due to human factors (mistakes in manufacturing, operation, etc.), this area is of critical importance to design reliability. Good references on this also appear in Dhillon,[32] pp. l30–l32, and McCormick.[40]

For the analysis conducted in section 7.2, the actual sources used were the following:

- IEEE reliability data for pumps, valves, and actuators.
- Shuttle integrated risk assessment (SIRA).
- SAIC STS risk assessment.
- Engineering judgment.
- Reliability data from the process industry.
- Rome Reliability Center database.

An example of the data provided for a 4-in. ball valve from these databases is presented in table 2. Included in this are brief descriptions of the type of valve actuation—electro-mechanical actuator (EMA), the size, a general description, and the failure estimates for composite and selected failure mode failure rates. This is about as good as it gets. Some of the data are traceable to its source—most of the process data are from the chemical industry, but much of the environment information is simply not available. Again, engineering judgment is a key part of any reliability estimation process.

One other caveat on the use of data from the data sources listed above is necessary. It is critical that as much information as possible be provided on the ultimate sources of the data and on the hardware systems listed in the data. Decisions to include or not include data in the analysis should be based on accurate information that is traceable to the source. Only through the use of this kind of design information can a good decision be made on the use of such information in reliability estimates. Certain data resources often do not list the source or claim the source as secret, making it very difficult for the individual who has to select the data for use. This is especially true on data provided by vendors. Vendor estimates of component failure rates are a key source of such data in aerospace applications. Visibility into this data is key for components and parts that have an active operational history or a strong pedigree.

Table 2. Failure rate quantification data example.

Number	Description	Size
V1	LO_2 Fill & Drain Valve (EMA)	4
V3	LH_2 Fill & Drain (EMA)	4
V4	GO_2 Vent Valve (EMA)	4
V10	GH_2 Vent Valve (EMA)	4

Description	Source	Composite (/HR)	Fail Open (/HR)	Fail Closed (/HR)	Fail to Contain (/HR)
(Lox or Fuel F&D)	SIRA		4.80E–07	5.30E–07	5.30E–07
(Valve, Summary & Electric Rotary Actuators)	Rome	5.10E–06			
(Composite, all process control valves)	Process Industy		3.00E–07	3.00E–07	1.00E–08
(Composite all electric motor valves)	IEEE	6.92E–05	3.12E–05	3.79E–05	1.00E–07
(2–4 in., electric, ball)	IEEE	3.00E–06			
Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions					
(Lox or Fuel F&D)	SIRA		8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy		5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Calculate Composites Using "OR" Logic					
		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(Lox or Fuel F&D)	SIRA	**2.56667E–07**	8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy	**1.01667E–07**	5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Calculate Averages and LN Averages Using a Weighting Factor of "1" for all Since They are Fairly Close					
Compare the Resulting Composites and Modes with the "OR" of the Modes					
		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(Lox or Fuel F&D)	SIRA	2.56667E–07	8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Averages		**2.64693E–06**	**1.77665E–06**	**2.14942E–06**	**3.55556E–08**
LN Averages		**6.62713E–07**	**2.75013E–07**	**3.03184E–07**	**1.34878E–08**

Composite of Modes	Delta %
3.96162E–06	**–49.66858**
5.91684E–07	**10.717817**

Using the LN Average and Average for 4 in. (Composite of Modes Matches the Actual Composite Best)
Calculate Average of the Composites to not Overemphasize the Significance of the Modes or the Actual Composite
Then use the Distribution of Modes LN Averages for Distrubuting This New Composite Number

(Lox or Fuel F&D)	SIRA	2.56667E–07	8.00E–08	8.83E–08	8.83E–08		
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07					
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09		
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08		
(2–4 in., electric, ball)	IEEE	5.00E–07					
Averages		2.64693E–06	1.77665E–06	2.14942E–06	3.55556E–08	3.96162E–06	–49.668581
LN Averages		6.62713E–07	2.75013E–07	3.03184E–07	1.34878E–08	5.91684E–07	10.717817
New Composite and Modes		**6.27199E–07**	**2.9152E–07**	**3.21382E–07**	**1.42974E–08**	**6.27199E–07**	
These Probabilities can then be Converted Back to Time to Failure Exponential Distributions and to Reliabilities							
New Composite and Modes		6.27199E–07	2.9152E–07	3.21382E–07	1.42974E–08		
LAMBDA (SEC)		**1.04533E–09**	**4.85866E–10**	**5.35636E–10**	**2.38289E–11**		
Reliability		**0.999999373**	**0.999999708**	**0.999999679**	**0.999999986**		

Design and environment details are critical for distinguishing between useful and inappropriate data sources. For example, data on a 4-in. stainless-steel ball valve operating in a cryogenic environment is far better information than generic ball valve data which does not specify design or environment information. Unfortunately, these types of data seldom exist. Note that the approach taken here is to use design, environment, and source information to filter the data under consideration. No attempt is made in this approach to adapt or provide quantification "scale factors" to apply to the failure rates to be used in the analysis. From the authors' point of view, the data are often too crude to be used in such a fashion. Already shaky confidence in the fidelity of the output would not be helped through the use of such scale factors.

6.3 Applicability of Data

As will become more evident in section 7.2, the application of reliability data to a systems risk quantification is much more of an art than a science. There is much engineering judgment that is at work here. The purpose of this discussion is to show the issues associated with system reliability quantification, "warts and all." No attempt is made to hide anything from the reader. In the authors' opinion, this is often a problem in systems reliability presentations—the quality, the transformations, and the filtering of the data are often hidden from the viewer, with emphasis on the statistical manipulations of the metrics.

Knowledge of the design and environment detail is critical to the assessment of applicability of the data. A list of desired data reflecting high to low applicability exists for aerospace systems reliability applications and looks like this:

1. Flight hardware, same/simulated environment—direct failure data.
2. Flight hardware, test environment—direct failure data.
3. Test hardware, simulated environment—direct failure data.
4. Test hardware, test environment—direct failure data.
5. Surrogate hardware, simulated or test environment—direct failure data.
6. Quality data (condition reports, preflight and postflight)—indirect data.

Of course, relative to parts and structure, tops on the list would actually be PDA type of information—information related to the actual "physics of failure." This is so infrequently available and oriented to structures and parts that it is not considered for this type of systems quantification. Therefore, top on the list presented is accurate data collected on the flight hardware in a space environment. Of course, such data also rarely exists for the reasons discussed in section 6.1. Good environment models reflecting the performance, thermal, stress, dynamics, etc. of the hardware are important in making the applicability judgment. If any of these data are collected from a reliability perspective, such as testing to failure, then it is of greater importance than just steady-state operation.

Another category of data often used in aerospace reliability estimation is not included in this list. This is the expert opinion or "Delphi" source of data.[41, 42] In general, this is not considered as much a data source as a last-ditch response to the problem of a total lack of data and a way in which to exercise engineering judgment. This also goes for techniques that combine actual data with expert opinion such as bayesian reliability. What is to be done when absolutely no data sources exist is a very difficult problem. In this discussion, it is generally assumed that some direct data source exists. Section 6.4 discusses the common use of an indirect data source, the UCR counts.

In aerospace, most data come from categories 4–6 above. Hardware is usually tested in a ground environment for steady-state operation. Surrogate data include other types of similar systems or components that exist in industry and can be considered as comparable. Extensive quality data often exist on launch vehicles (STS) and its applicability will be explored extensively in section 6.4.

A brief example of the use of surrogate data will illustrate the process and the problems of using surrogate data to make predictive quantitative reliability assessments. A current engine under development at MSFC uses an ablative nozzle and chamber (instead of being actively cooled, it erodes during use).

Other systems' ablative nozzles/chambers are considered as surrogate data providers. Table 3 presents a summary of the information collected on the surrogate systems. Two sets of data were collected. The first set reflects similar systems in solid rocket motors (SRM's). In this case, it is only the nozzle that is ablative. Table 3 lists some key design and environment parameters for each nozzle, such as material (carbon phenolic (Ca phen) or silica phenolic (Si phen)), burn time, and chamber pressure *(Pc)*. It should be noted that other design parameters not listed are also important and a case can be made that they should also be considered. For example, the type of solid propellant and its inherent abrasiveness could be considered a key parameter. The second set listed in table 3 is liquid fuel systems; these too have ablative nozzles only. For those, which are considered more comparable, the operational failure data have been collected and are presented. The difficulty in using these data is obvious; there are no failures—reliability engineers need failures for the reliability metric. Second, it is still a relatively small sample. Third, design parameters such as Isp and thrust are widely different.

Statistical manipulation will not clear up the difficulties in using the data in the first place. Since aerospace mechanical reliability analysis is more of an art than a science, masking the weakness in the data with statistical manipulation seems inappropriate. Data determined to be too weak should be discarded from consideration.

Table 3. Ablative nozzle/chamber surrogate data analysis.*

Nozzle (solids) or Nozzle and CC (liquids)	Material	Weight (lb)	Exit Diameter (in.)	Burn Time (sec)	Pc (psi)	Thrust (lb)	Flts	Succ
Solids								
Star 12A	Si Phen	10.8	4.6	7.5	1052			
TE–M–344	Si Phen	0.4	2.6	2.4	1230			
TE–M–345	Si Phen	2.4	4.9	20.5	565			
TE–M–416	Si Phen	17.2	8.4	Classified	Classified			
Star 26C	Si Phen	19.8	12.9	16.8	640			
Star 17A	Si Phen	10.3	13.8	19.4	670			
Harpoon	Si Phen	27	6.4	3	1838			
Star 13B	Si Phen	3.7	8	14.8	823			
Rem Pilot Veh	Si Phen	7.2	4.1	2.1	1076			
Star 24	Ca Phen	13.2	15.3	29.6	486			
Star 27	Ca Phen	20.5	19.5	33.5	529			
TE–M–640-4	Ca Phen	12.5	17.5	32	682			
Star 30E	Ca Phen	38.4	23.4	49	563			
Star 30BP	Ca Phen	34.5	23.4	54	515			
Star 48B–PAM STS	Ca Phen	83.5	25.9	83	576			
Star 48B–PAM Delta	Ca Phen	97.4	30.3	83	576			
Star 37XFP	Ca Phen	71.2	23.6	65.5	535			
Antares III	Ca Phen	65.5	29	45	712			
Star 37FM	Ca Phen	75.2	24.9	64	529			
Liquids (ablative cc, radiative nozzle)								
AJ10–137 (Apollo service module)	Ca Phen	450	98.4	750 (max)	100	21500	12	12
AJ10–138 (Titan III transtage)	Si Phen	140	47.3	500 (max)	108	8150	46	46
AJ10–118F (N–2 second stage—Japan)	Si Phen	750	60.3	500 (max)	102	10000	8	8
TR–201 (Delta second stage—one-piece unit)	Si Phen	220	56.5	340 (max)	103	9900	67	67
Fastrac 15:1 (ablative nozzle & cc)	Si Phen	310	33	150	633	60000		

* Provided by Thomas Byrd, TD51, NASA MSFC

Options to expand this ablative nozzle surrogate database include exploring international launch vehicles and engines. The former Soviet Union has such engines. However, a problem may exist in getting access to the data, especially detailed design data. A better option would be to obtain test data on such comparable systems, liquid and solid, especially nozzle tests. Dealing with the problem of no failures in the data could be met with a worst-case assumption that the next flight will be a failure. This leads, in our case, to a simple ratio of 133/134 or a 0.9925 probability of success. This value should be looked at qualitatively and as generally useful in a comparative sense to other similar systems. It would be concluded here that, from a historical perspective, there can be confidence in the reliability of such systems. This is, at least in part, what such analyses are driving for—an analysis of historical data that provides (or does not) a sense of confidence relative to what is being currently designed.

One final comment is in order. Much of the discussion presented here may appear to be negative to the discussion of quantitative systems reliability analyses such as the PRA. Rather, a reflection on the methods and techniques of PRA is accompanied by feelings of incipience. And while there are good and justifiable criticisms of this approach, there is simply nothing else offered as an alternative. The kind of physics of failure analysis useful at a material or part level is not extensible to a systems level. Thus, the conclusion is that we have to make due with what we have—hopefully evolving and developing it into a useful and credible evaluation technique. Section 7.2 provides a detailed discussion of a quantified data analysis conducted for a advanced reusable propulsion system. Given the discussion in this section, this analysis should be seen as one possible approach in attempting to meet the goal of good reliability estimation for a future system.

6.4 Indepth: Unsatisfactory Condition Reports and Failure Rate

As previously discussed, in aerospace studies there is an acute lack of data to support the characterization of the reliability of systems and subsystems. Ideally, these data would come from direct sources; e.g., at 58 sec into test No.12, component No.788 cracked due to overheating and caused the engine to shut down. Since these types of data are relatively rare, reliability estimation has tended to rely on indirect types of data. UCR's are one example of this type of indirect data, and they are perhaps the most frequently encountered source of data for the quantification of failure rates for aerospace hardware.

6.4.1 Introduction

If a problem is encountered during test, checkout, and inspection, a special form is filled out—a UCR form. This form has changed somewhat over the years but has ≈25 fields that deal with UCR number, part name and numbers, reference procedure, reported by, engine number, date, how detected, description of problem, remedial action, type of problem, etc. Often, not all fields are filled in. UCR's generally do include a listing of human factors and process problems. In a typical review of UCR's, a spreadsheet list of these data will often be provided by S&MA contractors and may appear something like this for a problem with an engine sensor:

UCR NO.	ENG	SENSOR LOCATION	FAIL DATE	PREM C/O	PROBLEM DESCRIPTION
A032367	2015	LPFTP SPEED	03/11/93	N	OPEN CIRC. POTTING CRACK CAUSED BREAK

In this case, a low-pressure, fuel turbopump speed sensor on a particular engine in 1993 had a wire breakage due to a potting crack but did not result in an engine premature cutoff. The discussion of UCR data is referring to this kind of information.

It is important to distinguish between the engineering use of UCR's and the statistical use of UCR's. The engineering use emphasizes the analysis of hardware problems based on a detailed individual look at the UCR information. The emphasis is on finding the cause of the problems on an individual basis, looking at the exact phenomenology. UCR's provide notification and traceability to design and process problems that need to be resolved. These problems may be related to the reliability of the system, but not necessarily so; thus, the use of UCR's is necessary and critical. This use is not drawn into question in this section.

The statistical use for reliability characterization, the topic of this section, uses the UCR counts to provide reliability estimates, relative to the system generating the UCR's or to another proposed system. Establishing a failure rate of 1/10,000 from UCR counts is an example of the statistical application to a quantitative reliability estimate. Causation is not important here and this calculation of failure rate is irrelevant to the engineering use of UCR's discussed in the previous paragraph. This discussion will focus on the statistical use of UCR's.

Most frequently, the UCR's are filtered to the system of interest with very early development data excluded (green run/acceptance/calibration tests). Tests and flight data are used since UCR's are generated in all cases. Although, in some cases, the UCR counts are used to support a direct reliability calculation; most often they serve as the basis for weights or allocations. For a new system, given or assuming an overall reliability and percentages associated with the different subsystems or components from a comparable system, reliability of components is generated. These numbers are based upon the basic allocation due to UCR counts, part count comparisons, predicted improvements, and expert opinion; then this is rolled up to a new overall system reliability number for the new or updated system.

Current efforts and a literature review have failed to show any persuasive connection between the indirect (UCR's) and direct evidence. This section attempts to identify any correlation between the UCR data and direct evidence. It attempts to do this by a top-down approach; i.e., a general discussion of the problem for the reader; a data analysis by looking at J–2 and SSME experience with UCR's; and a theoretical development of the problem.

6.4.2 Background

Estimating engine failure rate from a history of thousands of tests may seem a simple problem. The real problem is not "what has been," but "what is going to be." The problem may be more properly stated as, "based on a history with constantly changing engine configuration and test conditions, what are the failure odds on the first flight of the next engine off the production line?"

One approach is to run a large number of Monte Carlo replications on a full-blown computer simulation that expresses all engine "physics." This would be an ultimate system level PDA. It is unlikely that such a massive task has ever received serious consideration.

Another approach is to simply count tests and failures and use the Binomial equation to estimate engine failure rate at a confidence level. Sounds simple enough, but then you get into problems with which configurations and test conditions to include and how to "count" tests and failures. For example, from a risk point of view, two 20-sec tests may count the same as one 250-sec test (J–2 engine). The real problem is the number of engines required for a reasonable failure rate and confidence. If reasonable is defined as a failure rate ≤1/1,000 at a 90-percent confidence level, then we would need to test over 2,000 engines without a single failure.

Because of problems with using direct data, there have been attempts to use indirect evidence, such as the failure history of similar components. For example, one might contend that a valve is a valve. If so, then we could collect failure data on all valves and use fudge factors to correct that data to the complexity and environment of a specific engine valve.

For several years there have been attempts to use QC-type data, such as UCR's, to aid in estimating engine failure rates. For engine programs, such as the SSME, J–2, F–1, and the H–l, there may be a few hundred engine premature cutoffs, but thousands of QC defects. Hence, if some relation exists between engine failure and QC defects, QC defect data might be very useful in better understanding engine failure events.

Common sense suggests a relationship: a hardware design is defined by a set of drawings, procedures, and specifications. The design engineer, in effect, asserts that if these requirements are met, the hardware failure rate will be acceptable. Hence, it seems reasonable to assume that the hardware failure rate will be less acceptable if these requirements are not met. If the QC system is absolutely 100-percent effective, then only within-spec hardware will be installed in an engine and the hardware failure rate will, by definition, be acceptable. If the QC system is <100-percent effective, then some out-of-spec hardware will escape the QC system and be installed in an engine. In other words, failure rate would tend to increase as QC defect rate increases. A QC defect may not be a "real" problem, but symptomatic of a problem. The tacit assumption seems to be: "Where there's smoke, there's fire!"

Yet, there seems to be no persuasive study that shows a useful relation between QC-type data (defect rates and/or events) and engine failure data. It is easy to understand how some might draw false conclusions from historical data. For example, if the historical data selected for evaluation happens to be when both the engine cutoff rate and the inspection rate is nearly constant, then one might conclude that historical data show a relation between UCR's and engine cutoffs.

A "top-down" evaluation of SSME and J–2 engine data, which spans a significant change in engine cutoff rate, indicates that there is no empirical relation between the number of UCR's and the number of premature engine cutoffs. These studies indicate that the number of UCR's is driven primarily by the number and kind of inspections. The more you look, the more you find. Section 6.4.4 presents the results of this analysis.

More engine tests equal more cutoffs and inspections. More inspections equal more UCR's. Hence UCR's and cutoffs may tend to "travel" together because engine tests are common to both, but that does not mean cutoffs and UCR's are otherwise connected. This becomes obvious when the engine cutoff rate changes without a corresponding change in the UCR rate—or the reverse. After the STS *Challenger* accident,

many more UCR's were generated in the flights immediately after return to flight, before the number of UCR's returned to more typical levels. One suspects that sensitivity to any type of problem was very high after the *Challenger* accident, resulting in the drastically increased number of UCR's generated.

Some might contend that studies would show a useful relation between UCR's and engine failures, if the data had been correctly evaluated. "Correct" evaluations might include, for example, different ways of screening and trending the data. However, as discussed in section 6.4, analyses on data collected using several filtering techniques have not been successful in generating a relationship that is useful and consistent.

Although overwhelmed by other factors, a weak connection between UCR's and engine cutoffs should exist because:

- An engine that experiences a premature cutoff or a component with a history of problems may be subjected to more intense inspection.

- The failure mode that triggers a premature cutoff may, incidentally, damage other hardware—secondary failures or damage. More damage equals more UCR's.

Such UCR's may follow problems or our perception of problems, but are not very useful for predicting engine failure. One would have to assume that our perceptions are always correct and no corrective action was effective.

Historical data also show that some UCR's are nuisance reports. A nuisance UCR is defined as the same condition that is reported a number of times, in a short time period, without immediate and strong corrective action. In other words, the condition is tolerated, because immediate corrective action is not worth the trouble. In such cases, the importance of the condition reported is inversely related to the number of UCR's. If a condition is considered critical, strong and immediate corrective action may preclude recurrence. Hence, important conditions reported via UCR's tend to be comparatively rare events. Nevertheless, a large number of UCR repeats over an extended period may indicate a problem that is difficult to fix. The "problem" may be due to lack of process control and/or a design error.

Other problems show up frequently, are well known, but are disregarded before any analysis is done because they completely dominate the database entries. One such problem is evident with the STS thermal protection system (TPS)—dents and nicks lead to a very large number of UCR's. Another is cracks in welds in the pumps; they are known problems, there is no easy solution, they are noted every test and flight, but they are considered outliers in any analysis since they would completely dominate the failure allocation.

This points to the issue that a large number of UCR's per engine failure may be due to the damage caused by the malfunction of one failure mode, rather than the engine failure being the result of a large number of UCR conditions. If a large number of UCR events occurred before engine hot fire, then this may merely indicate that the QC system was doing its job of keeping bad hardware off the engine. Thus, there would be no necessary or consistent relation between pretest UCR's and engine failure rate.

A large number of UCR's for a particular component or failure mode may simply indicate that the problem reported is not a problem worth fixing, rather than a problem that is hard to fix. Perhaps, when these UCR's were generated, there were other problems that needed to be fixed that were a higher priority. If a UCR event is really important, it may be fixed immediately and thus never reoccur. In such a case, we may find that the significance of a UCR event is inversely related to the number of UCR's on such an event.

6.4.3 Practical Considerations

There are other concerns evident in the use of the UCR data. It is easy to see why their use is so attractive—at first glance, the large number of UCR's that exist would appear to lend themselves very well to statistical calculations of probabilities and confidences. However, with further scrutiny, other problems are evident.

The discussion so far has pointed to the notion that many different types of problems are noted on UCR's. Anything from cracks, loose parts, dings, to human factors are recorded. Thus, the database is oriented to safety, reliability, operations, and maintenance concerns. Sorting out just what relates to reliability is the challenge at hand. Several databases related to the SSME are kept at MSFC. It is illuminating to compare the entries. The first is the UCR database—over 7,000 SSME UCR's were recorded over a period of post-*Challenger* accident through 1995. A second database maintained at MSFC only records early cutoffs of engines during test and flight. One could assume that this database is more relevant to a reliability study—events that were serious enough to lead to an actual termination of a hot fire should be more applicable. Over the same time period, this database has ≈416 entries. Finally, another database is maintained at MSFC that is considered to be a major event database (generally considered to be actual hardware failures). In this database, all UCR's and reviews of early cuts are carefully scrutinized by a team of design and reliability engineers from NASA and Rocketdyne (the engine contractor) and only actual failures are listed. Over the same timeframe, this database contained 32 entries. One could use this database for analysis; however, the statistical nature of the data (large sample) has obviously been lost. This last database does not include human or processing errors.

Finally, there are the basic data recording problems. Often the engine test number is not recorded in a UCR. Hence, there is no way to link the two—critical for evaluating sequence and equivalent full duration (EFD) risk analysis.[43] Also, the error rates on recorded data, both within and between the UCR databases (MSFC and Rocketdyne) are high. Reconciliation of these errors would require a massive manual operation.

6.4.4 Data Analysis

For the following analysis, both SSME and J–2 historical data (UCR's and early cutoffs) were used. Figure 14 presents the cumulative UCR counts for several types of tests and components and premature test/flight engine cutoffs by time for the SSME. This figure does the best job of summarizing the overall problem in using UCR counts to calculate failure rates. This analysis included the use of 7,000^{+} UCR's collected on the SSME over a 20^{+}-yr period. During this period, there were ≈420 early cutoffs of test and flight engines.

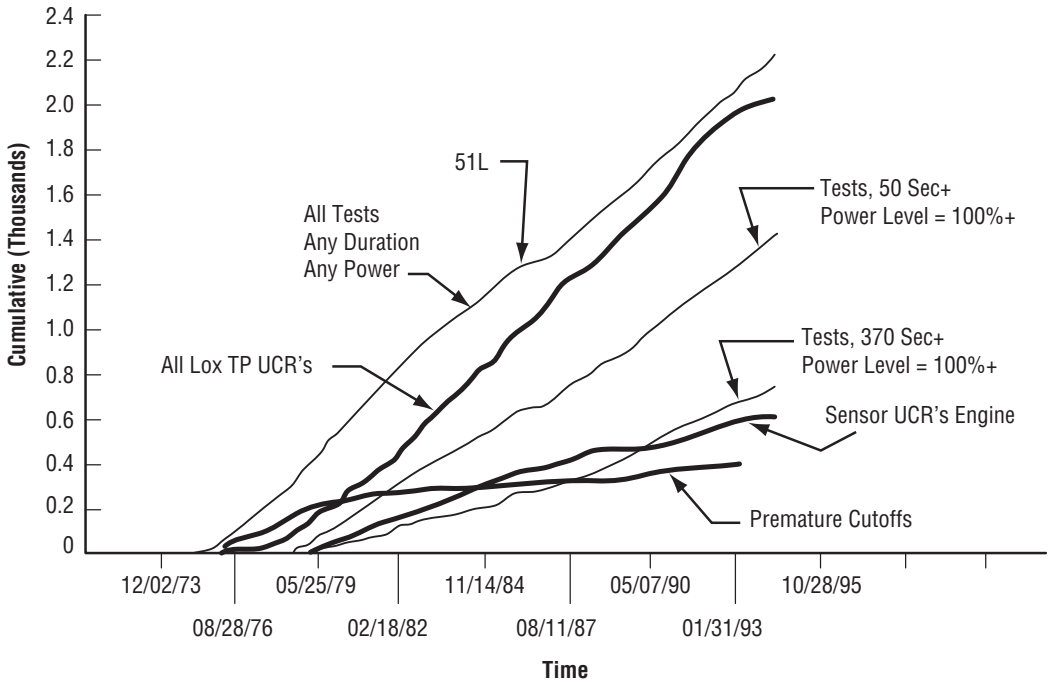


Figure 14. SSME UCR history.

From figure 14 it is apparent that the curve over time for the engine premature cutoffs rises for 2.5 yr or so and then begins to level off. This is what one would expect over the life of a program with extensive testing and analysis—problems are found through testing and solutions applied to the problems over time. This will reduce the number of problems experienced over time. In this case, it is assumed that premature cutoffs are more reflective of "true" failures of the system. Of course, the true reliability will never be known, but prior studies have shown a connection between the two. Since premature cutoffs tend to drive discrepancy data, the correlation between premature cuts and discrepancies will be higher than the correlation between "true" engine failures and discrepancies. In other words, if there is no connection between premature cutoffs and discrepancy data, then there cannot be a correlation between UCR data and "true" engine failure. Unfortunately, a proof of a connection between discrepancies and premature cutoffs is not necessarily proof of a connection between UCR's and "true" engine failure.

The rest of the lines on the graph in figure 14 reflect the different UCR count totals. The top line reflects the number of tests conducted (over 2,000) with a decline due to the *Challenger* accident (51L) shown. The accident resulted in no flights for over 2 yr and reduced testing, as reflected in this line. Other lines reflect specific component UCR's, such as lox turbopump and engine sensor UCR's. The other two reflect subsets of the total tests—those tests that ran longer than 50 sec and those that ran longer than 370 sec.

None of the UCR curves presented in figure 14 or, for that matter, those components not presented, contain the "knee" in the curve that is evident in the premature cutoff curve and that would be expected through the course of test and development of aerospace hardware. Also, there is no way to normalize the basically linear UCR curves to the basically nonlinear premature cutoff curve. This alone is strong evidence that there is no consistent or strong correlation between the two.

For the following analysis, the J–2 discrepancy data were used (≈5,000 entries). Also, ≈4,000 J–2 tests and flight data were available. In general, the discrepancy database was accepted at face value. This database was used, among other things, to develop a risk distribution equation that was used to normalize all tests to risk of a certain duration (i.e., 250 sec). The risk factor was applied to every test in the database to take out the effects of different planned test durations. For example, all else being equal, a J–2 engine test planned for 20 sec sees half the risk of a 250-sec test. A 500-sec test sees 1.2262 times the risk of a 250-sec test. The 20-sec tests were counted as 0.5 of an EFD[43] and the 500-sec test was counted as 1.2262 EFD, when full duration was defined as 250 sec.

Figure 15 presents the early J–2 engine cutoffs over the cumulative EFD for production (PROD) and research and development (R&D) engines. There were ≈150 production engines and 50 R&D engines. In general, the same "knee" in the curve that was presented for the SSME data exists for the J–2 data. After a certain period of time, problems found during testing are fixed and, over time, the incidence of problems diminishes. The slope of the curves after the "knee" is generally linear and similar for production and R&D engines.

Figure 16 presents the UCR history by early cutoffs and figure 17 presents the same by cumulative EFD. Notice that the production engine generates a mostly linear trend in figure 17 and the R&D generates a slowly curving trend without a noticeable "knee." Again, this is very similar to that of the SSME, however, R&D J–2 engines present a more nonlinear trend. The strongest effect in the data is the difference between production configuration engines and the R&D engines. The production engines experienced a much higher rate of discrepancies per cutoff than the R&D engines, even when both were experiencing a similar premature cutoff rate. It is suspected that this is because production engines were subjected to a much higher rate of inspections and "checkout" tests.

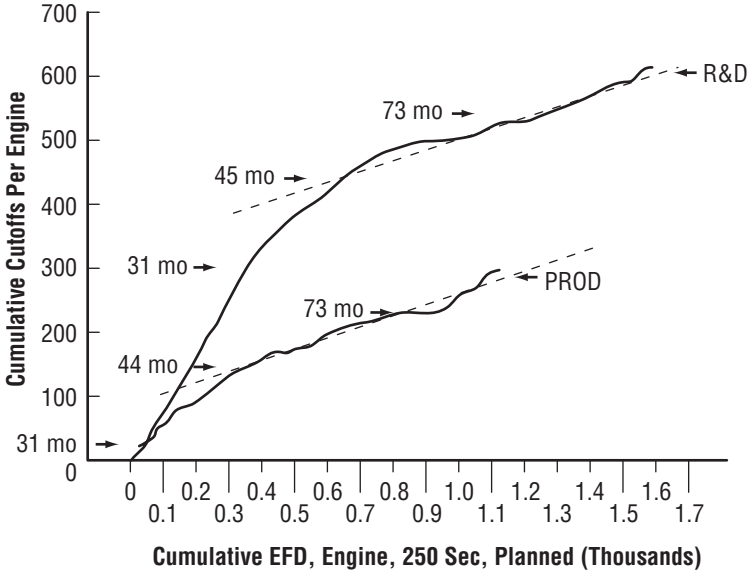


Figure 15. Early cutoffs for J–2 engine by cumulative EFD.

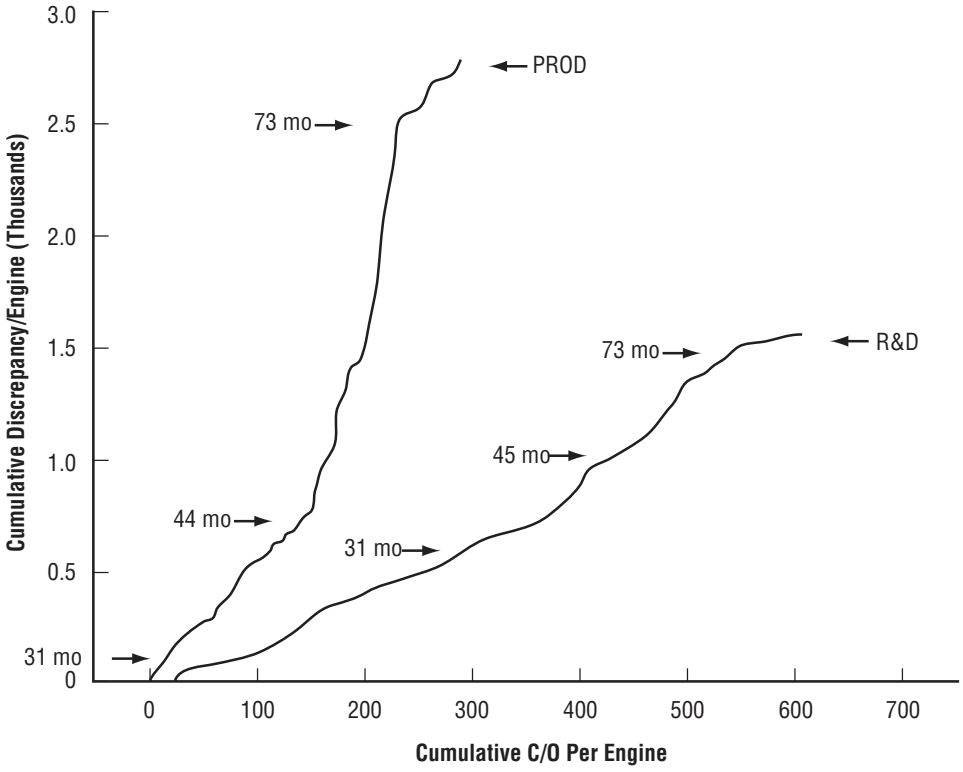


Figure 16. J–2 engine UCR's by cumulative cutoffs.

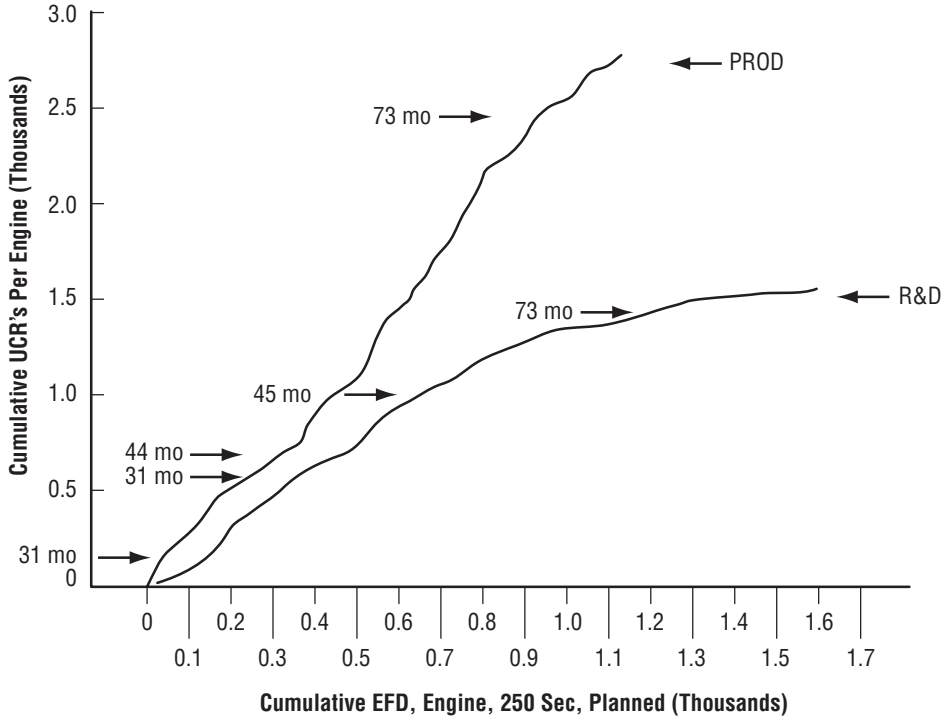


Figure 17. J–2 engine UCR's by cumulative EFD.

An observation can be made here that the trend of R&D UCR's appears to be closer to what is expected and is evident in the early cutoff curve than the trend of production UCR's. Philosophically, the process of collection of the two sets can be seen as very different. During R&D, the goal is developing a useful engine that operates correctly. One suspects that the emphasis is on actual problems that keep the engine from operating correctly, not on dings, dents, and other miscellaneous problems that would catch the attention of quality personnel inspecting production engines. On a production engine, the emphasis is on catching anything that can impact quality, safety, reliability, and maintenance, generating a much broader set of UCR's. Though this has not been fully investigated, perhaps some filter of R&D quality data could lead to a good dataset for reliability purposes.

Typically, a production engine slated for flight would follow this sequence of events:

1. First electrical and mechanical (E&M) checkout.
2. Engine acceptance tests.
3. Second E&M.
4. Receiving inspection at the site of "stage" acceptance test.
5. A "stagemate" inspection when the engine is installed.
6. A "prestatic" checkout before the first hot fire.
7. Stage acceptance hot fire.
8. A "poststatic" checkout after last stage hot fire.
9. A prelaunch checkout.

The R&D engine is not generally subjected to any of these tests and inspections. In the database of ≈4,000 tests and flights, not a single R&D engine was acceptance tested, nor did a manual scan of engine histories in a paper database reveal any E&M checkouts. Basically the R&D engine was subjected to some sort of inspection after every test. Inspection following a premature cutoff would be more intense than one following a successful test.

Figure 18 seeks to present an explanation for some of the upward changes in the lines for the cumulative UCR's. Cases of disassembly (Disassy) or overhaul of the engine relative to the timescale indicated in the figure have been labeled. The R&D engines late in the program were subjected to extensive overhauls and inspections. Coincidentally, the number of UCR's increased. Also, the early production engines were subject to the inspections and checkouts listed earlier, and coincidentally, the line changes accordingly. Upward movements seem to be roughly correlated to increased inspection opportunities: early for production engines and late for R&D engines.

This scenario seems to support the assumption that "the more you look, the more you find." It cannot be proven with the existing databases but how else can this data be explained? If there exists some basic and fundamental relation between UCR and engine premature cutoff data, then this relation would be a constant for any cutoff rate. In other words, a plot of discrepancies against cutoffs would not show a "knee." If this relation is constant over the life of the J–2 program, then a cumulative sum plot of R&D engine data and a plot of production engines should produce two straight lines that fall on top of each other.

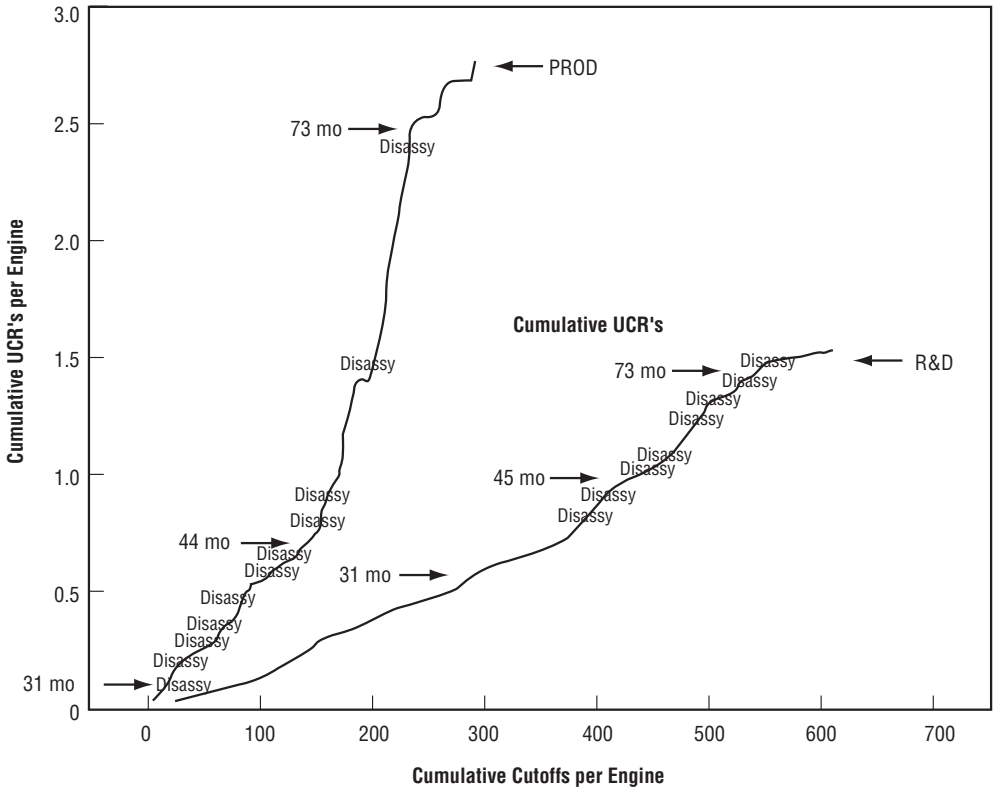


Figure 18. J–2 engine inspection opportunities.

6.4.5 Theoretical Considerations

Hidden failure modes are another source of misleading information. Hidden failure modes are those that may never fail because some other failure mode almost always fails first—one or more failure modes "hide" behind a primary mode.

1. Liquid rocket engines are fluid dynamic machines, hence the load at any one point in a fluid circuit may be highly correlated with all other points in the same fluid circuit.

2. There may be several different failure modes (and/or components) in the same fluid circuit. Because all these failure modes see a "common" load driver, then all these modes are correlated to some degree.

3. These failure modes will not have the same failure odds. One will be the "weak link." Figure 19 shows how this might look if all modes were normalized to a common load.

The primary failure mode is a "weak link" that is consistently weaker than other "links" in the same chain. Generally, the QC system does not know which mode (and/or component) is the primary one and which are "hidden" modes. Thus, the components with the hidden modes are subjected to the same QC procedures as the primary mode. These hidden modes may generate more UCR's than the primary mode, but make no significant contribution to the system failure rate.

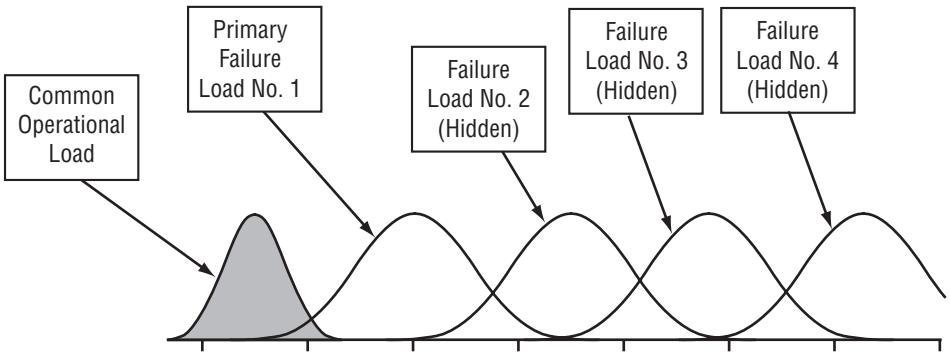


Figure 19. Hidden failure modes.

The "hidden" mode problem becomes apparent during engine development, when a design fix for one problem uncovers a new problem. For example, a design fix may move the primary mode in illustration No. 1 "off scale" to the right and "failure load No. 2" becomes the new primary mode. Near the end of the Apollo program, the engine program office (H–1, J–2, F–1, and RL10 engines) conducted a study of the F–1 and J–2 engine programs. This study indicated that ≈100 J–2 failure modes were found and fixed in ≈4,000 J–2 engine tests. Later, during the Shuttle program, a simple test-fail-fix computer model was built to provide some insight into the SSME development process. A number of different approaches were tested against the J–2 database. In a preliminary study, the best fit resulted with the assumption that the J–2 engine consisted of 30 primary independent modes with an infinite "stack" of hidden failure modes behind each of the 30 primary modes.

A better test-fail-fix model and more work may reveal a different number of primary modes and a different hidden mode structure, but a satisfactory data fit without some hidden mode assumption seems very unlikely. If we are willing to accept this preliminary study as a reasonable indicator, then one would have to conclude that most engine failure modes are hidden.

There is a theoretical relation between UCR-type data and engine failure rate, but it is not consistent. The relationship varies significantly from parameter to parameter, failure mode to failure mode, as a function of process shift type and the statistical properties of the parameters involved. In other words, it is not possible to develop a credible estimate of hardware failure rates by using QC defect data. The following is rationale supporting these assertions.

This rationale includes three limit conditions that would preclude a relationship between UCR's and engine failure. These conditions start with a load or stress that the hardware experiences and a corresponding load or stress required to break that hardware. If the condition required to break the hardware is called failure load or strength and the experienced condition is called operational load or stress, then the following is true.

The first limiting condition is when the dispersion of the operational parameter is much larger than the dispersion of the failure parameter, the QC system for the operational parameter is perfect, and the distribution of the failure parameter is well outside the QC spec limit for the operational parameter. Then, for all practical purposes, there will be no engine failures, regardless of the QC reject rate for the

operational parameter. For all practical purposes, the failure distribution is too far above the QC spec limit for a random failure load parameter to reach below the QC spec limit and no random operational load parameter can ever exceed the QC spec limit; therefore, no failures regardless of QC reject rate. In other words, data from the "fat" operational load distribution cannot reach the "thin" failure load distribution because of the perfect QC "fence" for the operational load. This is absolutely true if the standard deviation of the failure load distribution is zero. Hence, there is no correlation (see fig. 20).

Another limiting condition is the opposite of the preceding, as shown in figure 21. Namely, the dispersion of the strength (failure load) distribution is very large relative to the stress (operational load) distribution. In this case, the engine failure rate will be about the same regardless of the QC reject rate, even if the QC system is perfect. In this case, the failure distribution may be close enough for a random failure load parameter to reach below the QC spec limit, but because the standard deviation of the load parameter is so small, relative to the failure parameter, modifying the operational distribution by rejecting hardware will have very little effect on failure rate. This is absolutely true if the standard deviation of the operational load is zero. Hence, there would be no correlation between QC defect rate and engine failure rate (see fig. 21).

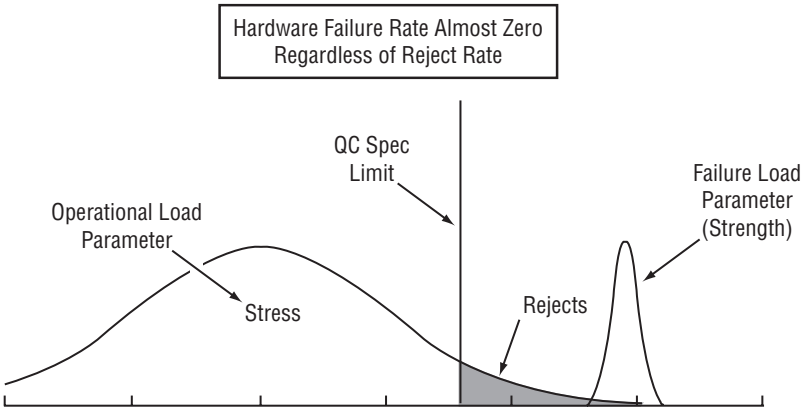


Figure 20. First limiting condition.

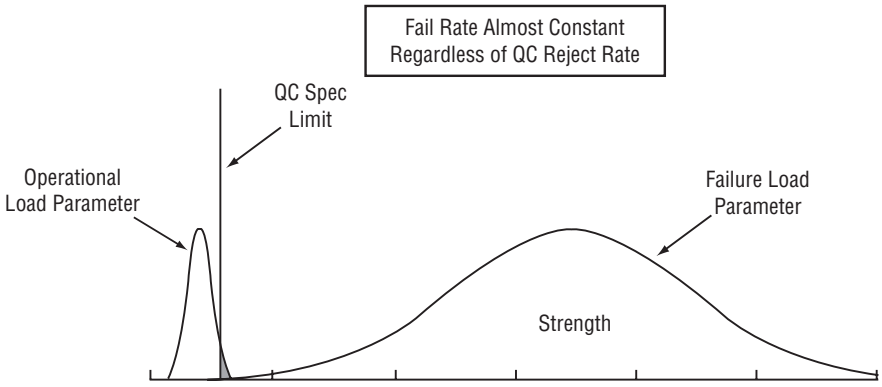


Figure 21. Second limiting condition.

The first limit condition is true because the QC system controls the major source of variability (operational load) of those parameters that drive engine failure rate. The second limit condition is true because the QC system does not control the major source of variability (failure load). The preceding illustrations are based on subjecting the operational load or load driver to inspection. Analogous conclusions would result if the failure load (strength) had been subjected to QC procedures.

Most of the real world exists somewhere between these two limits: the need is to investigate this region between these limits. Because failure parameters are difficult, if not impossible, to measure on an engine-by-engine or test-by-test basis, an operational parameter should be selected for study.

Operational load or stress parameters are easy to measure; therefore, they are the source of many UCR's. Operational load drivers include such parameters as wall thickness, diameter, pressure, and revolutions per minute (rpm). Some operational parameters may be measured several times a second during an engine test. Other operational parameters may be measured before and after each engine test. Failure load or strength parameters are difficult to measure. Most are "measured" indirectly, just once, by use of witness specimen "tag ends," hardness tests, or expensive test-to-failure sampling. Generally, there is an abundance of reasonably accurate operational measurements and a shortage of failure load measurements. The accuracy of the failure load measurement is not well known. Accurate failure load measurements may require a test setup that mimics the engine loads and environment very closely.

The operational parameter was selected for study because of the relative abundance and accuracy of data. The untruncated failure distribution may be viewed as the output of a QC system that "controls" the failure distribution but is not accurate enough to truncate it. The inclusion of truncated failure distributions would make failure rates relatively insensitive to UCR rates. Figure 22 depicts what might be expected if both distributions were truncated. The difference between the QC spec limit for the operational load parameter and the QC spec limit for the failure parameter might be determined by an SF or some other design criteria.

If the difference between the two QC spec limits happens to be

$$\text{DELTA QC SPEC} = 4.76 * [(\text{STD DEV OPS QC ERR})^2 + (\text{STD DEV FAIL QC ERR})^2] \ ,$$

then the maximum possible failure rate will be <1 out of 1 million, regardless of QC reject rate of either or both distributions. The reject rate for the load parameter might be 90 percent at the same time that failure parameter is experiencing a 90-percent reject rate, before the hardware failure rate would approach 1 out of 1 million. Under these circumstances, you would not expect many hardware failures in the lifetime of most engine programs, but a large number of UCR's might be generated. Since the standard deviation of measurement error tends to be much smaller than the standard deviation of the parameter being measured, then the difference in the QC spec limits could be quite small. A design based on such criteria (QC design margin) would be very robust with minimum performance impact. Although the QC design margin is not the usual design criteria, some failure modes may incidentally approximate this criteria. This is the third and ultimate limiting condition that would preclude a correlation between UCR's and engine failure. In the "real world," the failure distribution might be significantly truncated. If all other QC is ineffective, proof test may limit the failure load. Truncation of the failure distribution is, after all, the primary purpose of proof tests. The first two limiting conditions (figs. 20 and 21) are considered rare events, but this third limit condition may be fairly common.

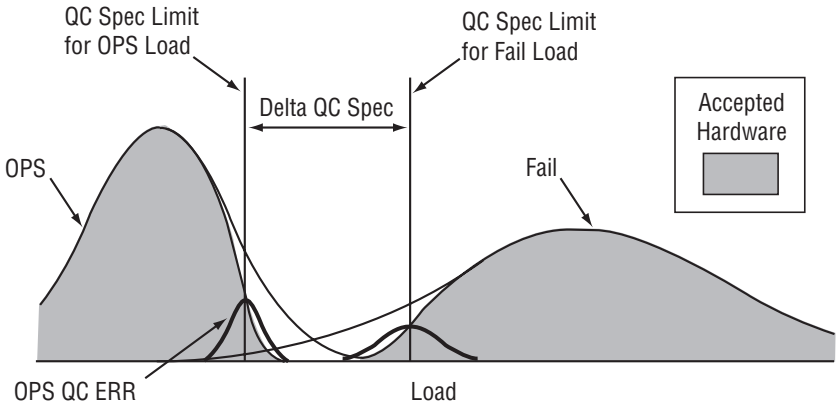


Figure 22. Third limiting condition.

This concludes the theoretical discussion of the relationship between UCR's and failure rate. An excellent reference that further studies this topic and expands this discussion to include stress/strength model development can be found in Lishman.[44] In this reference, the relation between UCR's and model development is examined in considerable detail. Models were developed to investigate two kinds of process shifts, four inspection scenarios, and a number of different input assumptions. This investigation revealed that the relation between UCR's and engine failure rate changed with any change in process shift, inspection scenario, input parameter, or defect rate. It is shown that many different engine failure rates are possible for a given UCR rate. All the conclusions presented here are supported in more detailed analysis contained in this reference.

6.4.6 Conclusion

The previous section discussed the statistical application of UCR counts to the calculation of quantitative failure rates. Again, this is carefully distinguished from the engineering use of UCR's—a necessary and critical function that identifies, traces, and attempts to solve individual design, hardware, and process problems. The conclusions reached here refer only to the statistical application of UCR counts to the generation of failure rates.

The "real world" is full of mixtures or distributions of process shifts. To use historical data for the construction of a "UCR versus failure rate" chart for a specific failure mode and inspection scenario, one would have to compare the UCR's from a specific primary load driver with the failure rate of the corresponding failure mode, when the UCR rate is changing and the failure load distribution is constant. If the failure distribution is changing as data are collected, it will not be known how much of the failure rate change is due to a change in UCR's and how much is due to change in the failure distribution—most of the time, little is known about the failure distribution. If the UCR rate is not changing, an empirically determination of how much the failure rate changes as a function of UCR rate cannot be made. Not only must the failure load be constant, but the failure rate of all other failure modes must be constant. If the failure rate for all other failure modes is changing as data are collected for the selected failure mode, then the selected failure mode's share of engine failures would also be changing. If the engine failure rate is scattered over 100 equal failure modes, then only 1 out of 100 engine failures would be due to the selected mode. If the

number of engine failure modes change or just become less equal, then the selected mode's share of engine failures will change. It might be very difficult to make any sense out of such data. It is difficult to imagine a "real world" where conditions required for a valid "UCR and failure rate" estimate would exist for sufficient time to collect enough data. Nevertheless, if these requirements are met, then one would have to repeat such a study for large sampling of different kinds of parameters and failure modes, before one could show empirical evidence of a universally consistent and useful relationship—if such exists.

This may explain why prior studies have failed to demonstrate a useful relation between UCR's and hardware failure. One must assume that efforts to use UCR's as some indicator of hardware failure rate are based more on faith than on fact. The preceding and Lishman~ show that the relation between UCR's and engine failures depends on many factors-three of which are inspection scenario, rejection rate per parameter, and "other" engine failure modes.

Finally, an area that needs to be explored more fully is the application of filtering techniques to UCR data—some combination of direct and filtered indirect (UCR) data may provide the best quantitative estimate of reliability. Perhaps a collection of filtered UCR's could provide accurate fault initiator information with test data providing the information on performance and environments that are not well understood.

7. APPLICATIONS

7.1 Qualitative Analysis Example

A recent program which required and benefited from qualitative design reliability analysis was the main propulsion system (MPS) design effort for the X–34 technology demonstration program.[45] This program will demonstrate enabling technologies supporting development of future RLV's, using a high-altitude demonstration vehicle. This vehicle, after being carried to an altitude of 38,000 ft by an L–l0l1 carrier jet and released, follows a flight profile which will demonstrate various technologies. The X–34 demonstration vehicle is being developed by Orbital, with the vehicle MPS design provided by an MSFC-led design team.

In order to meet X–34 system reliability requirements, Orbital levied a qualitative reliability requirement on the MSFC-provided MPS design: the MPS shall be two-fault tolerant to a catastrophic event while the vehicle is attached to the carrier, during the vehicle drop transient after release from the carrier, and during vehicle ground operations. The MPS is deemed two-fault tolerant to a catastrophic event if there are no credible, potentially catastrophic failure modes resulting from less than three concurrent initiating faults. This requirement defined a catastrophic failure mode as one which could cause loss of human life. The MPS design fault-tolerance analysis was performed by the MSFC Propulsion Systems Analysis Branch in cooperation with the MSFC S&MA office and MPS design team engineers.

7.1.1 X–34 MPS Design Fault Tolerance Analysis Task Structure and Interfaces

The X–34 MPS design fault-tolerance analysis task was structured as illustrated in figure 23. The analysis was led by MSFC's Propulsion Systems Analysis Branch and supported by the MPS design team. The design team provided engineering expertise in establishing failure and operational assumptions, as well as providing necessary engineering analysis and modeling support. The results of the analysis were reviewed and coordinated with the S&MA office.

Since the MPS design fault-tolerance analysis was performed in parallel to the MPS design effort, the analysis results were able to influence design modifications, instrumentation and control definitions, operations timeline and limits, and operational procedures.

Once the MPS design was complete, any design failure modes which did not meet the two-fault tolerance requirement were to be reviewed by the S&MA office, Orbital, and the MPS design team. The S&MA office would then submit design fault-tolerance requirement waiver requests to Orbital.

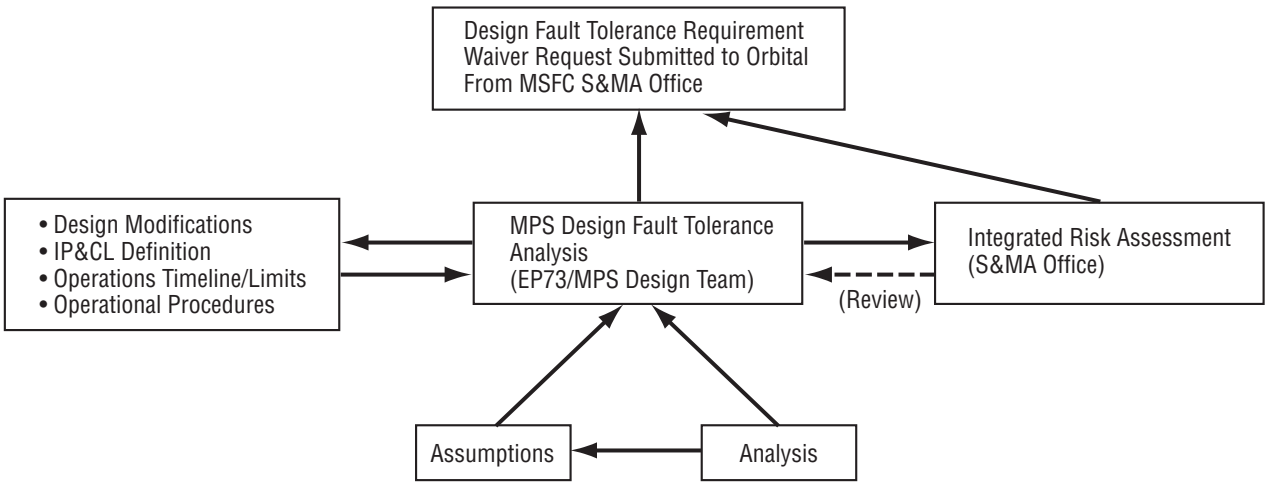


Figure 23. X–34 MPS design fault tolerance analysis structure and interfaces.

7.1.2 MPS Design Fault Propagation Modeling and Evaluation

The analysis of the MPS design fault tolerance required modeling and evaluating the predicted propagation paths of credible system faults. These system faults were identified in the X–34 MPS Integrated Risk Assessment Report (rev. B) NAS8–0364[46] and were selected per the MPS design fault tolerance analysis ground rules and scope.

The propagation of the credible system faults to a given system state were modeled as digraphs using the FEAS–M software described in section 5 as the modeling development environment. The following is a description of the digraph symbols used in fault propagation model segments presented in this report.

Initiating faults of modeled propagation are presented in figure 24, final system state of modeled propagation is presented in figure 25, and propagation paths between states appear in figure 26.

Figure 24. Example initiating faults.

Figure 25. Example final system state.

Figure 26. Example propagation path.

Logical “OR” gate is represented in figure 27. The logical value of target node is TRUE if any immediately preceding node is TRUE.

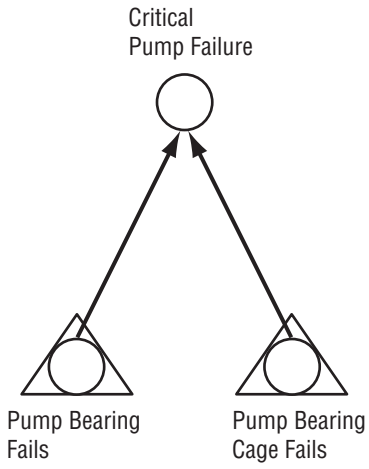


Figure 27. Example logical “OR” gate.

Logical “AND” gate is represented in figure 28. The logical value of target node is TRUE if all immediately preceding nodes are TRUE.

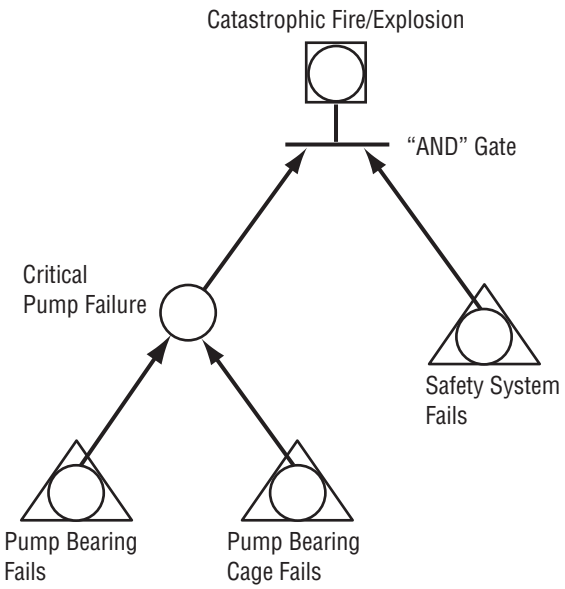


Figure 28. Example digraph.

The above digraph shows that if either a pump bearing or a pump bearing cage fails, a critical pump failure occurs. However, both a critical pump failure and a safety system failure to mitigate the failure must occur for a catastrophic fire/explosion to occur. In this example, if the safety system attempts to mitigate the catastrophic failure but is unable to do so in time, the safety system is deemed to have failed.

7.1.3 Influence of Design Fault Tolerance Analysis on the MPS Design

Most MPS design decisions required a balancing of contradictory design factors, not only of cost, performance, and weight, but also of factors related to safety and fault tolerance. A design decision made to eliminate one failure mode many times created another failure mode. Therefore, design decisions many times were based on eliminating high-risk failure modes while acquiring lower risk failure modes. The design fault-tolerance analysis provided a valuable input to these design decisions by assessing the credible failure modes considering the failure environment, failure propagation rate, phase, and mitigation provisions. The following are a few of the many influences the design fault-tolerance analysis had on the MPS design.

7.1.3.1 Placement of the IPS purge supply line pressure relief valve. The turbopump of the Fastrac engine used in the X–34 vehicle consists of an integrated package of an RP–l pump, a lox pump, and a hot-gas turbine. Propellants within the RP–1 and lox pumps are separated by an interpropellant seal (IPS) to which the MPS supplies a helium purge. This purge maintains propellant separation by providing a positive pressure in the IPS interseal cavity. If this purge is interrupted while propellants are present in the pumps, the propellants may mix, causing a fire or explosion. Therefore, the fault-tolerance analysis evaluated failure scenarios which may lead to a loss of this IPS purge.

Figure 29 shows the original design of the MPS IPS purge supply line. Helium stored on the X–34 vehicle at 5,000 psia is regulated to 750 psia by the IPS purge supply line regulator before passing through the IPS purge supply line isolation valve. The purge supply line pressure relief valve is placed downstream of the isolation valve, allowing the vehicle purge and pneumatic supply to be isolated from the pressure relief valve in the event that the pressure relief valve fails open.

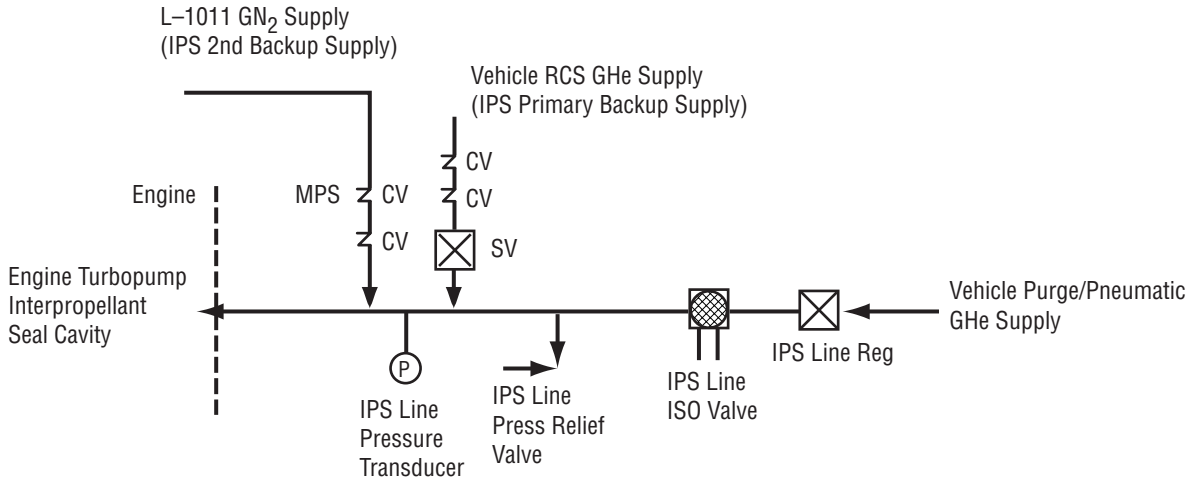


Figure 29. MPS IPS purge supply line, original design.

An assumption of the original IPS purge supply line design was that maintaining pneumatic capability during the captive/carry mission phase was critical and that maintaining an IPS purge was not critical. However, further evaluation deemed that the IPS purge was critical during part of the captive/carry phase since propellants are present in the engine turbopump several minutes prior to the release of the vehicle. Figure 30 illustrates the failure scenario for the original design in the event that the IPS pressure relief valve fails open. Once the safety system detects that the pressure relief valve has failed open, the IPS supply line isolation valve is closed, maintaining pressure in the pneumatic system upstream of the isolation valve. Since the pressure relief is failed open downstream of the isolation valve, IPS purge supply is lost, which is considered a catastrophic event. Therefore, the pressure relief valve was moved to upstream of the isolation valve and two IPS supply backup sources were added to the supply line, one from the vehicle reaction control system (RCS) and the other from the L–l011 carrier nitrogen supply. The revised design of the PS supply line is illustrated in figure 31.

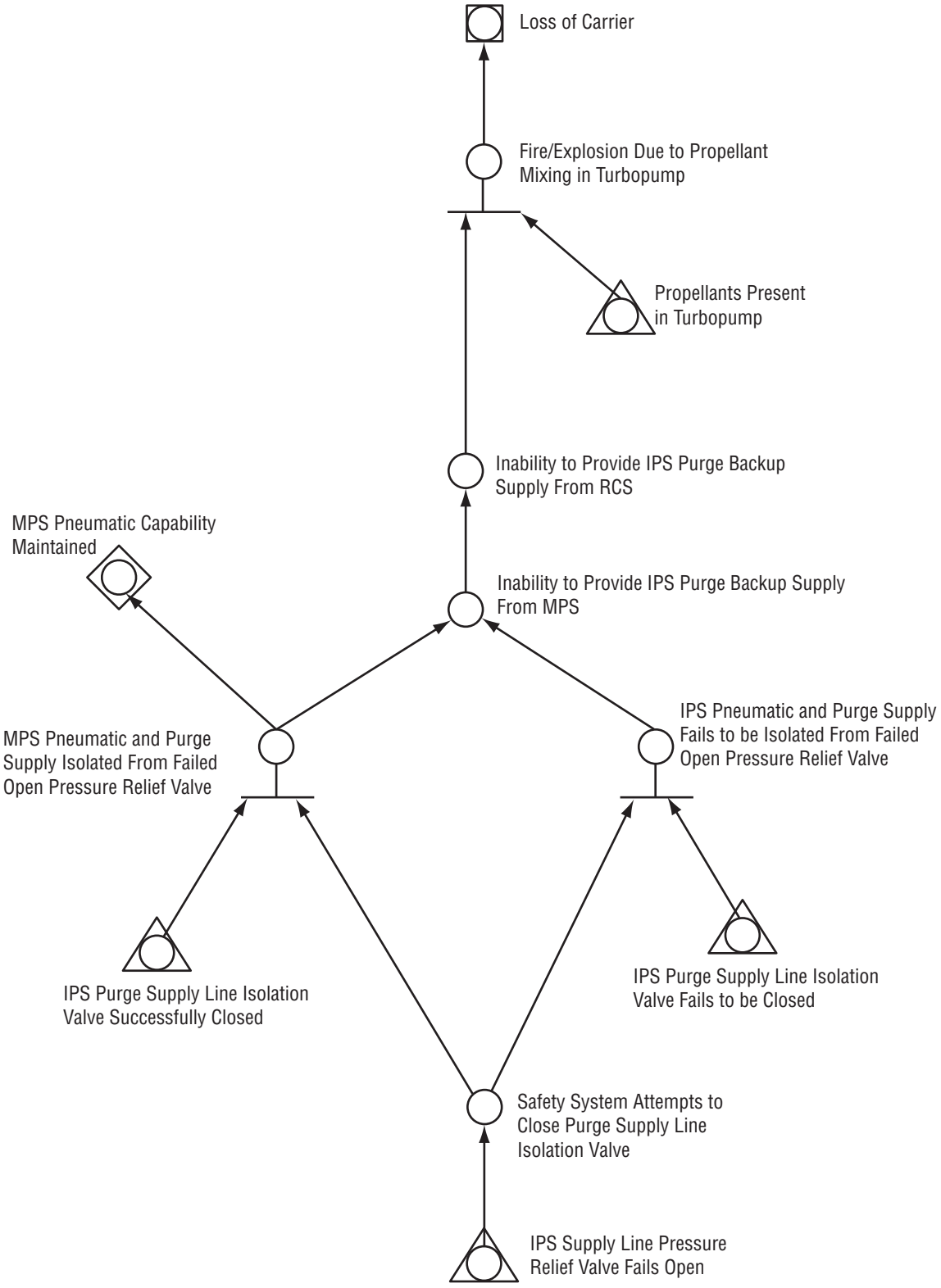


Figure 30. Original MPS IPS purge supply line design failure scenario.

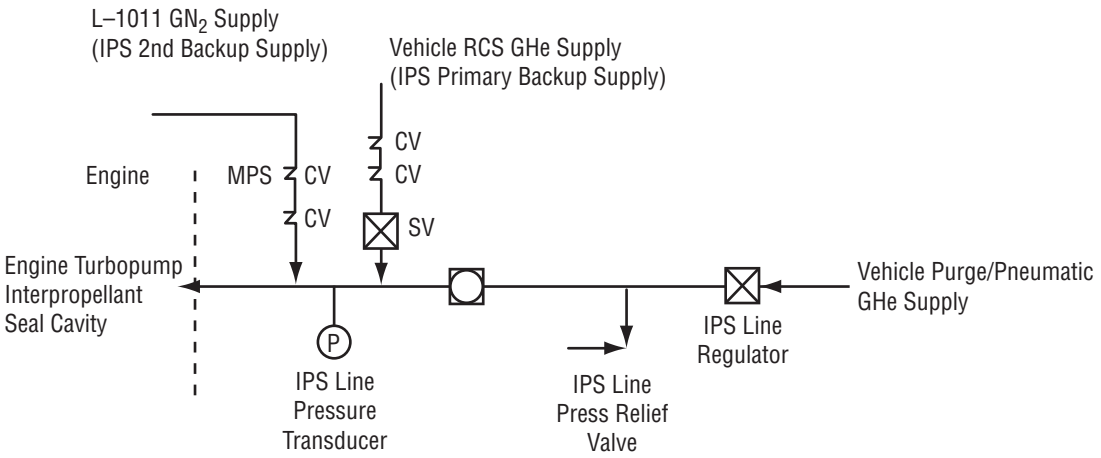


Figure 31. MPS IPS purge supply line, revised design.

The failure scenario of the revised IPS purge supply line design, in that the IPS pressure relief valve fails open (fig. 32) is shown to be two-fault tolerant to a catastrophic event. For this failure scenario, if the IPS pressure relief valve is failed open, the safety system attempts to close the IPS purge supply line isolation valve. Once the isolation valve is closed, the IPS backup purge supplies may be initiated. This design change, however, creates the failure scenario that the pneumatic system is disabled if the pressure relief valve fails open. Therefore, the consequences of a disabled pneumatic system were evaluated and deemed acceptable prior to vehicle release.

A detailed description of the analysis supporting IPS purge supply fault-tolerance analysis may be found in appendix C.

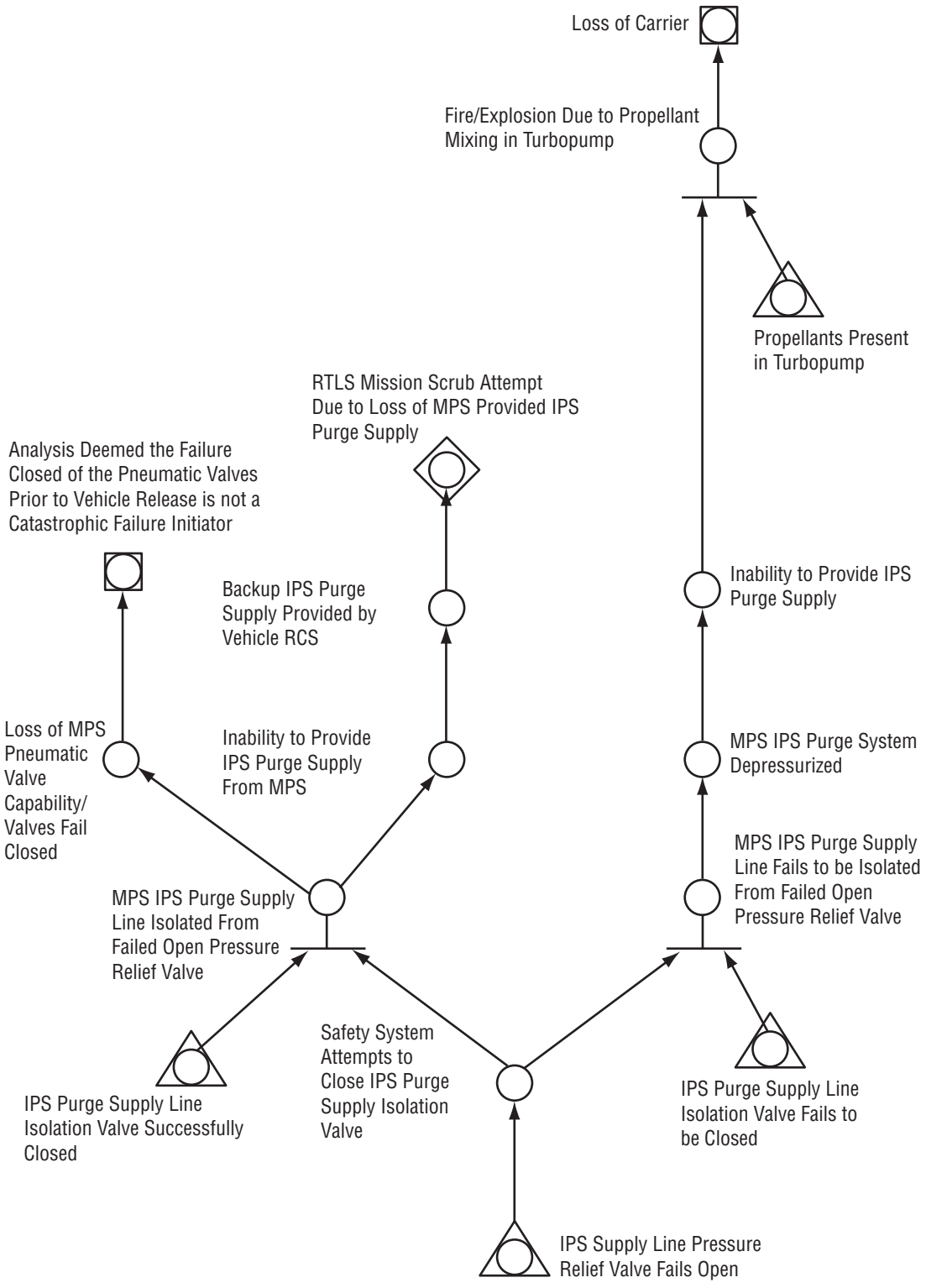


Figure 32. Revised MPS IPS purge supply line design failure scenario.

7.1.3.2 Determination of the IPS purge line isolation valve maximum response time requirement. Another failure path shown in figure 32 begins from the point that the safety system attempts to close the IPS purge supply line isolation valve. This failure path occurs if the IPS purge supply line isolation valve does not close in time to allow the backup purge supply system to adequately maintain the IPS purge before a propellant mixing in the IPS cavity occurs. This failure path was evaluated by determining how soon after a loss of IPS purge pressure at the engine interface that propellants could mix in the IPS cavity due to a loss of purge pressure. This analysis determined the IPS purge line isolation valve maximum response time requirement that was levied onto Orbital.

7.1.3.3 Definition of lox tank pressurization line check valve redundancy. A segment of the original MPS tank pressurization system design is shown in figure 33. Pressurant gas from MPS helium tanks is split—one leg supplies helium to the lox tank and the other supplies helium to the RP–1 tank. Pressurization supply in each leg flows through parallel servo valves, then through two check valves in series. The failure scenarios resulting from check valve leakage in the original design of the lox pressurization leg show that leakage through both check valves would expose the lox pressurization servo valves to lox. Since the seal specifications of these servo valves do not require tolerance to lox exposure, exposure of the seals to lox may result in seal damage and possible leakage of lox into the pressurization manifold and vehicle MPS compartment. Since only two failures, leakage of the check valves, were required to initiate this potentially catastrophic event, an additional check valve was added to the lox pressurization leg to make this failure mode compliant with the two-fault tolerance requirement. This revised design is illustrated in figure 34.

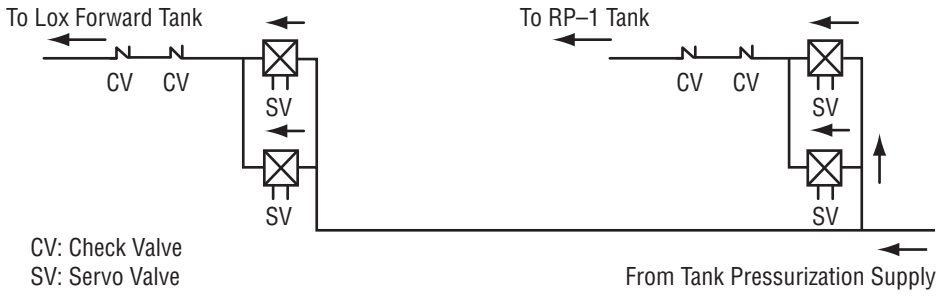


Figure 33. X–34 MPS tank presurization system (segment), original design.

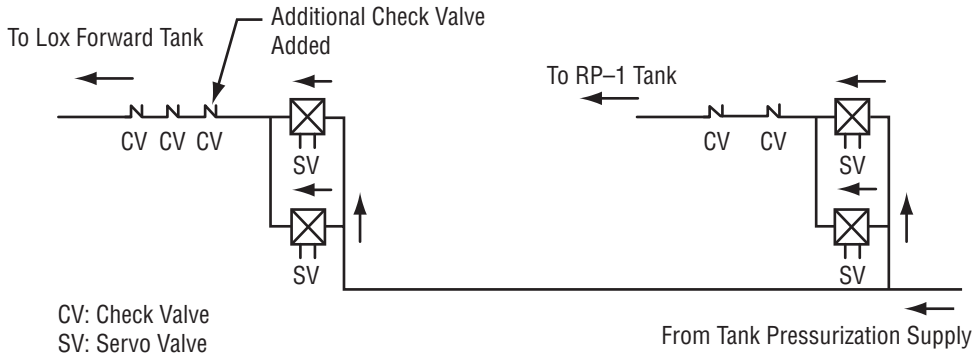


Figure 34. X–34 MPS tank pressurization system (segment), revised design.

7.1.3.4 Definition MPS valve latching and fail-safe requirements. The fail-safe position of the MPS pneumatic actuated valves was specified to be closed in the event that pneumatic supply to the valves was lost. Fault-tolerance analysis was performed for the failure modes initiated by a loss of supply pressure to the pneumatic actuated valves, resulting in these valves closing. The fault-tolerance analysis deemed that loss of supply pressure to the pneumatic actuated valves, given the fail-to-close specification, was two-fault tolerant, thereby verifying the valve fail-safe specification.

Controlling the MPS pneumatic/purge supply is electrically actuated servo valves. The fault-tolerance analysis deemed that these servo valves should have the capability to be latched in a specified position. By specifying that the servo valves supplying the IPS purge pressure be locked in the open position, IPS purge pressure would be maintained in the event that vehicle power is lost or the vehicle flight controller issues erroneous valve commands.

7.1.4 X–34 MPS Design Two-Fault Tolerance Requirement Compliance

The MPS was designed to avoid failure modes that were less than two-fault tolerant to a catastrophic event. However, six failure modes that were less than two-fault tolerant were not able to be designed out, given the cost and weight requirements levied on the MPS. These noncompliant failure modes were evaluated and deemed by the S&MA office, the MPS design team, and Orbital as acceptable, given their low probability of occurrence.

7.2 Quantitative Analysis Example

This analysis was conducted for two launch vehicle programs—the X–33 and the X–34. Again, as stated in section 6.3, this should be considered only a candidate approach to the very difficult problem of quantitative systems reliability estimation. These studies emphasized the quantification of the reliability of MPS components. Quantification of example MPS components will be provided here and in appendix D. This section focuses on the actual generation of the failure mode failure rates—the most critical part of the whole analysis. These rates would then be used at the leaf node level in a model of the failure logic in FEAS–M. The complete model setup in FEAS–M is not presented here.

For MPS components, four surrogate data sources have been identified. These are previous aerospace studies utilizing various sources of data, process industry commercial data sources, other industry data from the Institute for Electrical and Electronics Engineers (IEEE), and the nonelectronic parts reliability data from the Rome Reliability Analysis Center. Each of these sources will be discussed in turn.

Three significant aerospace studies of this type that have been conducted in recent years are the SIRA, Galileo mission risk assessment, and the Space Shuttle probabilistic risk assessment (SSPRA). These analyses used extensive amounts of engineering judgment and “Delphi” (or expert opinion) techniques in order to develop component failure rates for mechanical hardware. Their quantification techniques are well documented for purposes of qualifying and reinterpreting the information for use in new programs. This information is provided at the component failure mode level as needed. One of the most useful attributes of this information is the ability of the analyst to use the relative allocation or distribution of failure modes for application to components for which only component failure information is available. These data are generally considered to be limited in applicability due to the high degree of interpretation and engineering judgment required for converting this information into meaningful reliability numbers.

"Loss Prevention in the Process Industries"[36] provides little information on aerospace hardware, but does provide nuclear industry and chemical process industry data at both the component and component failure mode level. Much interpretation and engineering judgment is required for its application to a launch vehicle. As above, this information is best used for relative comparisons to the limited aerospace information and for assisting in "allocating" the component failures to the component failure mode level. Of particular usefulness is the information on nuclear industry components that are subjected to approximately the same design quality as aerospace hardware. This would tend to provide for higher reliability. Some of this design robustness may be counteracted by the significant use of redundancy in the nuclear industry. This would tend to reduce the individual component reliabilities in exchange for cost considerations. Much the same as in the aerospace industry, some modes cannot be tolerated in the nuclear industry, such as leakage. These modes should tend to be "designed out" by a similar philosophy. This source does not provide significant information on the relationship between fluids environments or component size and reliability.

Although a bit dated, "Reliability Data for Pumps and Drives, Valve Actuators, and Valves" is an excellent source for establishing allocations between failure modes. This source is useful, as stated above for "Loss Prevention ...," but also includes information on the relationship between fluids environments, component size, type of component (e.g., pneumatic valve actuators versus hydraulic actuators, and butterfly valves versus ball valves) and reliability. One caveat—some of these data appear to be from "Delphi" sources, an expert opinion approach discussed earlier.

"Nonelectronic Parts Reliability Data," NPRD–9l/95, Rome Reliability Analysis Center: This source has the most applicable, actual data for a limited number of components. Although this source provides data that are quite incomplete and have a limited traceable pedigree, it does provide fairly reasonable data for the aerospace industry. This is primarily in the area of small valves, pumps, and electromechanical components. The most significant drawback to this source is the lack of component failure mode information.

One possible method for quantifying MPS reliability models is to use a combination of the available information listed above. Exactly how the information is used is heavily dependent on the type of component. The different sources provide varying quantities and qualities of information on the different types of components. Each source must be examined for the proper type of information for the particular component of interest. A weighting factor can be used to reflect the perceived validity of the numbers.

Following is a step-by-step approach for a propulsion systems application, illustrated with the actual quantitative derivation presented here and in appendix D for a set of MPS components:

1. Use the existing aerospace applications as the baseline. Identify differences between them and the hardware for which the quantification is desired. Using pedigree information provided in these efforts and engineering judgment, establish a weighting factor which reflects the validity of the numbers to the hardware being analyzed. If the numbers are fairly close, the weighting factor will be set at 1.

2. Establish the closest applicable information (preferably nuclear industry components of similar size) provided in the process industry text. Compare this information with the numbers established in 1. If within an order of magnitude, use these numbers to add validity to the defense of the generated numbers. Set the weighting factor accordingly and apply.

3. Use the nonelectronics parts reliability database in the same way as the process industry data except compare the aerospace information to the nuclear industry data within this document. Compare the nuclear industry numbers with the numbers from the process industry data. If all three are reasonably similar, the nuclear industry data may be useful in adding additional validity to the final reliability number. If significantly different, adjust the weighting factors accordingly or discard the nuclear industry information. Compare the NPRD information with the SIRA/Galileo/SSPRA numbers. Adjust weighting factors accordingly.

4. Conduct the same process using the IEEE information. In addition, compare the "failure mode" level numbers provided in the SIRA/Galileo/SSPRA with the same from the IEEE. If close, this agreement can be used to assist the defense of the distribution of failure between modes. Apply weighting factors accordingly.

5. Based on the information available from each of the sources, it may be desirable to establish upper, most likely, and lower failure numbers. Establish a composite failure number for all modes for both log average and average calculations. Compare these averages; they should be close. Do the same with the distribution of failure numbers. The numbers for the averaged individual modes should combine by "OR" gates to the composites. Any data that are significantly outside of this baseline should receive a very low weighting factor or be eliminated as a source.

Following this process, failure rates have been calculated for a set of MPS components. One valve example is presented here and the rest are presented in appendix D. This section concludes with a discussion of the comparability of the rates generated by this method and those available through various other methods.

Table 4 presents the results of this analysis for a 4-in. EMA valve. These failure rate estimates are actually tagged by number to valves on the schematics used in our FEAS–M modeling tool and in the design drawings. Starting with the data sources listed, composite and failure mode rates are presented, where available. These values are first adjusted assuming a 600-sec mission and an exponential distribution. Composite values are then calculated using boolean "OR" logic. From this, averages and log averages are calculated and used to generate a new composite value that is then transformed back to time-to-failure composite and failure mode values. In this case, the composite mission reliability for this 4-in. EMA valve is 0.999999373. Analyses for other selected MPS components appear in appendix D. In several of those cases, very little data are available from the sources listed. Estimates were made based on the data available.

Table 4. EMA 4-in. valve failure rate quantification.

Number	Description	Size
V1	LO_2 Fill & Drain Valve (EMA)	4
V3	LH_2 Fill & Drain (EMA)	4
V4	GO_2 Vent Valve (EMA)	4
V10	GH_2 Vent Valve (EMA)	4

Description	Source	Composite (/HR)	Fail Open (/HR)	Fail Closed (/HR)	Fail to Contain (/HR)
(Lox or Fuel F&D)	SIRA		4.80E–07	5.30E–07	5.30E–07
(Valve, Summary & Electric Rotary Actuators)	Rome	5.10E–06			
(Composite, all process control valves)	Process Industy		3.00E–07	3.00E–07	1.00E–08
(Composite all electric motor valves)	IEEE	6.92E–05	3.12E–05	3.79E–05	1.00E–07
(2–4 in., electric, ball)	IEEE	3.00E–06			
Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions					
(Lox or Fuel F&D)	SIRA		8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy		5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Calculate Composites Using "OR" Logic					
		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(Lox or Fuel F&D)	SIRA	**2.56667E–07**	8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy	**1.01667E–07**	5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Calculate Averages and LN Averages Using a Weighting Factor of "1" for all Since They are Fairly Close					
Compare the Resulting Composites and Modes with the "OR" of the Modes					
		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(Lox or Fuel F&D)	SIRA	2.56667E–07	8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Averages		**2.64693E–06**	**1.77665E–06**	**2.14942E–06**	**3.55556E–08**
LN Averages		**6.62713E–07**	**2.75013E–07**	**3.03184E–07**	**1.34878E–08**
Using the LN Average and Average for 4 in. (Composite of Modes Matches the Actual Composite Best)					
Calculate Average of the Composites to not Overemphasize the Significance of the Modes or the Actual Composite					
Then use the Distribution of Modes LN Averages for Distrubuting This New Composite Number					
(Lox or Fuel F&D)	SIRA	2.56667E–07	8.00E–08	8.83E–08	8.83E–08
(Valve, Summary & Electric Rotary Actuators)	Rome	8.50E–07			
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09
(Composite all electric motor valves)	IEEE	1.15E–05	5.20E–06	6.31E–06	1.67E–08
(2–4 in., electric, ball)	IEEE	5.00E–07			
Averages		2.64693E–06	1.77665E–06	2.14942E–06	3.55556E–08
LN Averages		6.62713E–07	2.75013E–07	3.03184E–07	1.34878E–08
New Composite and Modes		**6.27199E–07**	**2.9152E–07**	**3.21382E–07**	**1.42974E–08**
These Probabilities can then be Converted Back to Time to Failure Exponential Distributions and to Reliabilities					
New Composite and Modes		6.27199E–07	2.9152E–07	3.21382E–07	1.42974E–08
LAMBDA (SEC)		**1.04533E–09**	**4.85866E–10**	**5.35636E–10**	**2.38289E–11**
Reliability		**0.999999373**	**0.999999708**	**0.999999679**	**0.999999986**

Composite of Modes	Delta %
3.96162E–06	**–49.66858**
5.91684E–07	**10.717817**
3.96162E–06	–49.668581
5.91684E–07	10.717817
6.27199E–07	

In comparison of these calculated numbers with other direct aerospace sources (SIRA, SAIC PRA), the composite values are generally of rough order of merit similar. For example, the composite for a 4-in. EMA valve was calculated here to be 6.3E–07. The composite from the SAIC PRA was 2.17E–07, at least a similar order of magnitude. Again, the SAIC PRA often depended upon the "Delphi" technique for quantification (expert opinion). Our technique used, for the most part, all field data that were available. Other composite values for the components were generally the same order of magnitude as other aerospace sources. On the other hand, failure mode failure rate estimates were often quite different. For example, for a 4-in. EMA valve fails-to-open value, it was calculated here to be 3.2E–07. From the SIRA, it is calculated at 3.3E–05. This is a crude formulation at best. Certainly, these values should not be considered as absolute measures of failure rates.

In conclusion, this is a subjective way of establishing failure rates and requires a significant amount of engineering judgment. Thus, the numbers are only slightly better than any one particular source. It is considered to be better in that the influence of factors outside the "old" hardware design (the unknown or not considered modes) are considered in the "new" hardware numbers. This method is considered better than the use of quality data to derive such numbers in that it emphasizes actual field data. Also, this approach uses design information as much as possible—a key difference from other methods. Finally, if significant agreement is found between all sources, defense of the numbers is much easier.

It should be noted that the numbers developed by this method do not represent "predicted" reliability, but are for purposes of establishing an approximate distribution of failures between failure modes and component failures. These numbers should obviously be considered "ball park." If upper, likely, and lower numbers are developed, an estimated range could be quoted. It would also be beneficial to compare these numbers to the numbers of other "experts" in the hardware and reliability analysis business, as is done here. It is realized that the above method is the approximate equivalent of "Delphi" techniques, but is heavily founded on actual hardware data versus pure engineering judgment. It is also considered much better than the use of quality data in the calculation of failure rates. Based on work presented in section 6.4, it appears that there is no relationship between UCR count and failure rate. The numbers presented from the application of this approach should be considered rough order of merit and used only in trades and relative design comparisons.

The numbers generated in this effort provide the component failure mode failure rate information. These are placed at the leaf node levels in the FEAS–M model and are then used to generate (propagate up) intermediate and top-level probabilities. These values are generated relative to the failure propagation logic that would exist in a FEAS–M model.

8. CONCLUSIONS

This TP has been, in effect, a summary of the design reliability activities of a propulsion system team for the past several years. As such, it was set up to accomplish several goals. The first goal was to outline the role of reliability in a design program (sec. 4). Design reliability is viewed as a core design activity of equal importance to performance, schedule, and cost. A comprehensive design reliability program must be in place at the outset of any launch vehicle development program. Primary reliability engineering is to be accomplished by the design engineers using effective reliability models and tools and practical design criteria with assistance from the cognizant reliability group.

A second goal stresses the importance of reliability modeling and the use of metrics. A tool to support model development and analysis was developed and discussed at length (sec. 5). In order for reliability to be taken seriously, it must be on an equal footing with performance analysis. For this to happen, there needs to be high-fidelity model input into design decisions. A step was taken in this TP to present a model and an analysis approach that makes such input more feasible.

A third goal involves the need to stress the importance of the qualitative type of analysis (secs. 4 and 7). Looking at a design in "failure space" is an important mindset and is critically important to any design process. Much can be determined in such an analysis that not only affects the reliability and safety of the system being designed, but the cost of the system as well. Designers must be involved in this since the level of detail is critical to the quality of output necessary to support design decisions. Also, models used in an example qualitative analysis were presented.

A fourth goal involves an extensive discussion of the use of reliability data in quantitative types of analysis (secs. 3, 6, and 7). The sources, quality, and applicability of data available to the reliability engineer were discussed at length and an example provided of such an analysis. The general conclusion was to make the best of a bad situation by using as much operational data as possible. In general, only data that clearly points to hardware reliability problems should be used. This favors the use of direct over indirect failure data, even if the direct is for surrogate systems and indirect exists for the actual system. Caveats were placed on the use of UCR-type data, data with no traceable pedigree, and analyses that generate "absolute" measures of reliability. With the use of qualitative and relative quantitative analyses, good comparisons between concepts and systems can be effectively supported. In such analyses, assumptions and data sources should be explicitly listed so that the designers can make an informed decision relative to the quality of the data and the fidelity of the analysis. The process that the reliability engineer takes to provide reliability inputs to the designer must be visible (as is so often not the case); any weakness of the data must be acknowledged upfront so the designer knows the fidelity of the analysis output. Also, comments in section 6 were directed at ways to effectively model human factor issues. These must be included in design analyses, as this will likely affect any conclusions or reliability estimations.

Finally, several points should be made regarding the future of the design reliability discipline. Section 4.2 emphasizes what should be obvious: the main purpose of the discipline of design reliability is for ensuring the design of reliable hardware. One of the criticisms of the reliability discipline is that it is

very manpower intensive and time consuming relative to a rather low fidelity product. This is a just criticism and reflects also that current design reliability input does not often impact the course of the design. What is needed in this view is design criteria—standards that directly impact the design. The design criteria should support the design process in a way that designers are familiar with. Section 4.2 discusses such design criteria and derives them such that they fit the traditional design process. It takes a probabilistic approach but evolves the results back to a deterministic application so that typical design methodologies can incorporate them with minimal impact. This section also scratches the surface on another area that should impact reliability but typically does not—the use of effective QC techniques to ensure the selection of reliable hardware. Much work still needs to be done in these areas.

One last comment about future direction. The design reliability discipline seems ripe for the development of new metrics and new approaches for ensuring reliability such that the traditional problems with data, which are not likely to go away, can be overcome. The search is on for new metrics linking reliability and performance. One view is that reliability is actually the consistency in the variability of some performance parameter. That is, reliability is how well the performance parameter stays within the acceptable performance variability (or range) over time. This is a potentially fruitful area for the exploration and development of new metrics. As it stands now, the fidelity of the design reliability analysis will always seem to be severely limited by the profound lack of data relative to the preferred metric, *R*. Thus, the discipline should engage in a search for new ideas, new directions, and certainly new metrics. Perhaps what is needed for reliability is a new metric that is comparable to the metrics of thrust or Isp for engine performance—characteristics that are meaningful, easily measurable, and can be updated after each significant event.

APPENDIX A—Selected Topics

This section provides extensive detailed information on other key topics in the field of design reliability. A general design criteria concept is presented in section A. 1, then possible simplifications are discussed in sections A.1.1 through A.1.4. Section A.2 explores the critical relationship between QC and design and section A.3 provides a brief discussion of reliability verification.

A.1 General Design Criteria

The recommended design criteria are based on the theory of PDA. This method is also referred to as "stress/strength" or "applied stress/resistive stress" analysis in many texts. PDA is viewed by many engineers as extremely resource intensive. This view is due to the many thousands of failure mechanisms contained in most designs. Although many think there are thousands, perhaps millions, of failure mechanisms in a reusable rocket engine, actually there are only three "mechanical" failure mechanisms: low-cycle fatigue (LCF), high-cycle fatigue (HCF), and wear. In some cases, these could be consolidated into one mechanism, since they all are a form of fatigue. PDA requires the statistical characterization of the load, or "stress;" the capabilities, or "strength;" and any correlation between. A comparison of the stress and strength distributions, with proper accounting for correlation, allows the calculation of reliability due to a particular failure mechanism.

The loads, or "stress," consist of pressures, temperatures, and dynamics, and their prediction uncertainty. The strength consists of material fatigue properties, material property measurement uncertainties, and stress analysis tool uncertainties. If these can be properly characterized, the PDA problem may be reduced to a deterministic analysis. Efforts to characterize materials strength are well on their way. Characterization of the design tool uncertainties is not. The biggest problem with the current and past material properties characterization is the way the information is presented to the designer; usually as 2- or 3-σ minimums. However, this information should be presented as the mean and sigma as a minimum, or, in the best case, as statistical distributions. Then, based on the type of distribution, design criteria can be established based on the type of material, process, desired reliability, and any other factors which affect the type and shape of the strength distribution.

The most difficult part is characterizing the analytical tools that are used to predict the loads and strength. Many assumptions will be required and many "detail" part and assembly level tests will be necessary to validate these assumptions, thus validating the reliability prediction.

Of course, it would be very inefficient to do a detail PDA on every piece part or have design criteria for each piece part. This could be overcome by grouping the types of hardware into categories and establishing criteria for each category, much the same as for SF's. The grouping may allow a single criterion to be developed for a given material and analysis method used for rotating hardware (HCF). Another grouping may allow the same for a pressure vessel (LCF). A third grouping may allow a criterion for the material

in a wear application. Undoubtedly, others will be required. These simple PDA-based design criteria will result in the ability to make more credible failure rate predictions or vice versa. This is an oversimplification, but it is much better than using SF's, and it reduces the amount of effort required in comparison to detail PDA's of every piece part.

This concept may appear to be neglecting the cumulative damage aspects of changing stress fields and proper cycle counting. For a real reusable system with very low mission-to-mission environmental changes, the stress and strength should not change significantly in a random fashion, but rather in some determinable fatigue/wear-related pattern. Significant changes only occur as the parts wear/fatigue. This allows correlation between the stress and strength to be determined and analyzed/designed into the hardware with proper design criteria considerations. Due to the competing stress and strength characteristics associated with the high-power densities required for space flight, the criteria cannot be readily met in many cases. These are referred to as "rock-and-a-hard-place" problems. In these cases, detail PDA will be used if practical design alternatives cannot be developed without significant programmatic cost/schedule impacts/risk.

Other, possibly significant, failure modes which are not addressed above include misinspection, misassembly, and other human factor effects.

With the proper characterization of the stress and strength drivers, this methodology could possibly be greatly simplified. Some of these simplifications are discussed further in the following subsections.

A.1.1 Some Practical Considerations

For any complex system, design engineers must investigate, at a detail level, a very large number of component-specific design failure modes. All of these design failure modes cannot be simultaneously incorporated into a single Monte Carlo model because there is not a large enough knowledge base, nor computer, and it would take forever to run enough replications.

To make this Monte Carlo practical, the methods must be simplified. In the early design and reliability allocation phases, 30 or so failure modes that are the primary drivers of dry weight could be selected. The rationale for selecting the best reasonable number is dependent upon the criticality and degree of independence of the failure modes. The rationale for failure mode selection would be dependent upon analysis of risk, functionality, cost, common failure mechanisms, and dry weight. Dry weight would be a key factor due to its direct impact on vehicle performance. After selection of the primary modes, all other failure modes are designed to more conservative design criteria so that they can essentially be ignored. Since it is not practical to build a complete model, a strong combination of engineering judgment and knowledge of probabilistic theory must be used to decide how conservative the criteria should be for the secondary failure modes.

A larger risk is taken on the heavy items because of the direct tradeoff with payload. There is little point in taking a big risk on items with an insignificant payload impact. A tradeoff of dry weight versus failure rate was selected for illustration purposes, because it is a simple tradeoff for many failure modes, and it provides a direct connection between failure rate and payload. Many other tradeoffs must be considered in subsequent and more mature models.

The secondary failure modes could be treated the same way as the total system. Namely, failure modes that represent the primary drivers for a given subsystem are selected and everything else within that subsystem is conservatively designed so that they can be safely ignored in the subsystem model. One primary exception to this approach is the "rock-and-a-hard-place" problem.

Many design engineers have a limited knowledge of probabilistic approaches to design. The probabilistic design criteria to be imposed on the designers should represent the least possible change to their traditional methods. If all hardware is to meet the design goals and requirements, then methods that all design engineers understand are needed. If the appropriate knowledge of probability and statistics cannot be converted into physical design criteria that any design engineer can use, the analysis is of limited value.

This strategy advocates the use of probabilistic methods and some reasonable worst-case assumptions to derive design allowables, which when used in standard engineering models would result in a failure rate equal to or less than some specified value. The methodology for doing this would be delivered to the engineers in terms of tables, simple equations, and/or simple desktop computer programs.

The price of this simplification may be a larger-than-desired degree of conservatism or robustness for some failure modes. If the failure mode is a major dry weight driver, or falls into the "rock-and-a-hard-place" category, it may be worthwhile to apply a more sophisticated method. Even then, the more sophisticated method (e.g., a high-fidelity Monte Carlo model) would not be practical until well past the initial trade studies and preliminary design iterations. Once this attempt to be exact is made, much effort may be expended in continually updating the Monte Carlo model and the hardware design parameters as the total system evolves.

A.1.2 Simplified Criteria Considerations

The probabilistic approach to design is usually based on some variation of a stress/strength-type model instead of an SF. The design criteria is usually expressed in number of standard deviations (a 6-σ safety index) or some probability (99.9999 percent).

Some propose an analytical propagation-of-error method for estimating the statistical properties of the strength and stress distributions. Others advocate a brute-force Monte Carlo approach. There are pros and cons to both, but neither approach is well suited for use by the typical design engineer, especially during the preliminary design phase. All proponents of these methods seem to agree that a complete and exact solution of a complex system is not possible. The tendency is to limit analysis to a few critical design failure modes and/or make so many simplifying assumptions that it becomes difficult to decide whether the result is optimistic or pessimistic.

If the probabilistic method can be used for just a few design failure modes, then it might be desirable to select those that promise the maximum potential for performance improvement, cost savings, and/or failure rate reduction. If it is desirable to attract more investment, then the selection would lean toward high-profile items, such as turbine blades instead of nuts and bolts. Since there is no strict criteria for selecting such modes, the selection will be based on engineering judgment. If there were strict criteria for selecting failure modes that need help, such modes would not exist, since the problem would be known before-the-fact and designed out.

If an investment is made in probabilistic design of just the "high-profile" failure modes, the system failure rate is likely to be driven by the vast majority of the "low-profile" failure modes. If so, then the investment in probabilistic design was made only to find that the operational system failure rate is not that much better. Therefore, a reasonable and economical method of addressing all failure modes must be developed.

The traditional, nonprobabilistic approach is to use various SF's and depend on the QC system, checkout procedures, proof tests, malfunction warning systems, and acceptance tests (hereafter referred to inclusively as the QC system) to ensure reliability once the system is "debugged." Generally, it takes thousands of tests and many years to "debug" a system.

Some contend that the problems merely reflect that the hardware is always at the "leading edge" of technology. Others contend that there will always be problems. There are simply some things that cannot be controlled nor predicted. Still others are quick to point out that hardware that never fails is too heavy to fly. They are all basically correct. It is suggested that, in the past, the primary reason for being less than successful is that there is, for all practical purposes, no worthwhile functional relationship between typical design requirements/criteria and failure rate. For example, there is no way to construct a rigorous failure rate estimate for any given hardware design failure mode, much less a viable system estimate, based on any typical contract end item (CEI) specification requirements. CEI specification requirements are not derived from a system analysis that, in effect, says that for a given system failure rate, these requirements must be met. Basically, the same design criteria are applied to everything.

Some SF's have pedigrees going back to Saturn V. If these SF's were good enough for Saturn V why are there still hardware problems? Since Saturn V, the industry has made significant improvements in engineering, QC, and process control. If these SF's provided sufficient design margin, then, with these new improvements, the current hardware should never fail. The same SF's have been applied to everything, regardless of complexity. Common sense seems to suggest the need for a bigger SF for complex items than for simple items. SF's have been used in aerospace engineering for a long time. Yet, it is easily shown that there is no useful or consistent relation between a SF and hardware failure rates.

A.1.3 Safety Factors and Safety Index

Referring to figure 35, the traditional SF can be expressed as:

$$SF = (AVG_{fail} - K_z * SIG_{fail})/(AVG_{ops} + K_z * SIG_{ops}) \ , \qquad (1)$$

where

AVG_{fail} = average failure load
AVG_{ops} = average operational load
SIG_{fail} = standard deviation of the failure load
SIG_{ops} = standard deviation of the operational load
K_z = a baseline K factor which would be used if an infinite sample size existed. The traditional K_z for material properties has been "A" basis per MIL–HDBK–5F[48] (2.326 for a normal distribution). For the load distribution, K_z traditionally varies between 2 and 4.

Referring to figures 35 and 36, Z may be expressed as:

$$Z = (AVG_{fail} - AVG_{ops})/(SIG_{fail}^2 + SIG_{ops}^2 - 2 * RHO_{fail,\ ops} * SIG_{fail} * SIG_{ops})^{0.5} \ , \qquad (2)$$

where

Z = the average difference between the failure and operational load, divided by the standard deviation of that difference.

$RHO_{fail,\ ops}$ = The correlation between the failure load and the operational load, and $-1 \leq RHO_{fail,\ ops} < 1$.

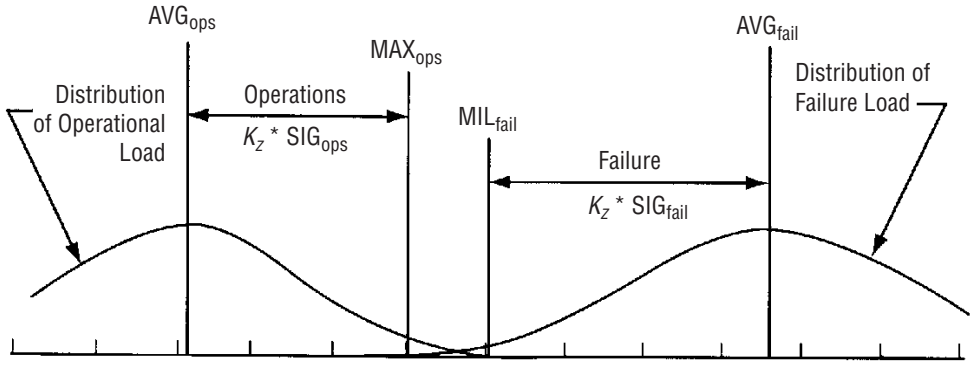


Figure 35. Derivation of traditional SF.

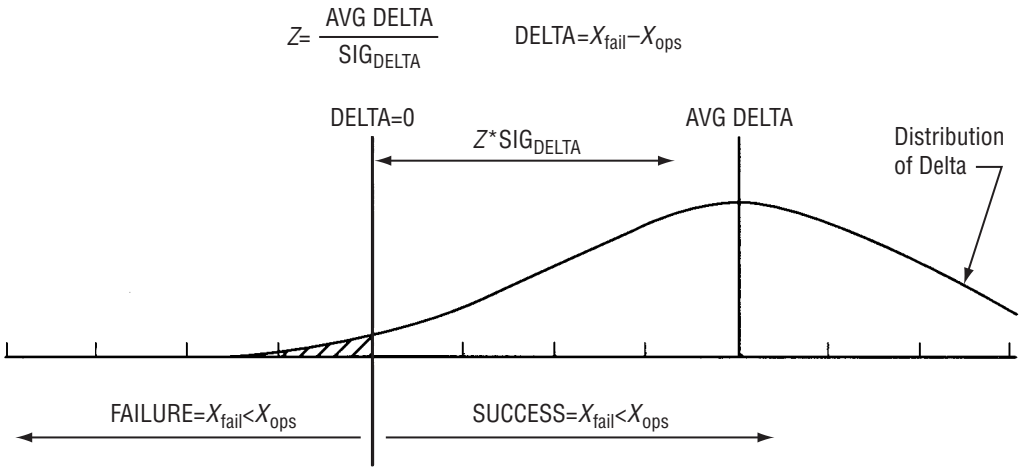


Figure 36. Derivation of Z.

The value of ($RHO_{fail, ops}$) tends to be negative. For example, the load capacity of a journal bearing increases as the rpm increases, but in a hypothetical hardware application, a load increase will cause an rpm decrease. Therefore, journal bearing failure load is negatively correlated to the operational load. It is suspected that a high percentage of failure modes are affected by a similar problem. Another example would be the *Pc* and failure pressure of the Shuttle's SRM's. If the SRM's run at higher than average *Pc,* then it flies faster than average and sees higher than average flight and heating loads, thereby reducing its capability to contain the *Pc.* Hence, the SRM's operational pressure is negatively correlated with the chamber's failure pressure.

Figures 37 and 38 show that *Z,* and hence the failure rate, may vary widely as the coefficient of variation *(CV_o)* of the operational load (standard deviation/average) varies for different SF's. Also, it can be shown that *Z* varies widely as the ratio of the two standard deviations *(C_{ss}* = SIG_{fail}/SIG_{ops}) vary. This relationship is developed by substituting equation (1) into equation (2) with = $RHO_{fail, ops}$ = –1 and both K_z's set equal to each other. Equation (3) is solved for *Z* in terms of SF, resulting in:

$$Z = [(SF–1)/CV_o + K_z(SF + C_{ss})]/(C_{ss} + 1) \ . \tag{3}$$

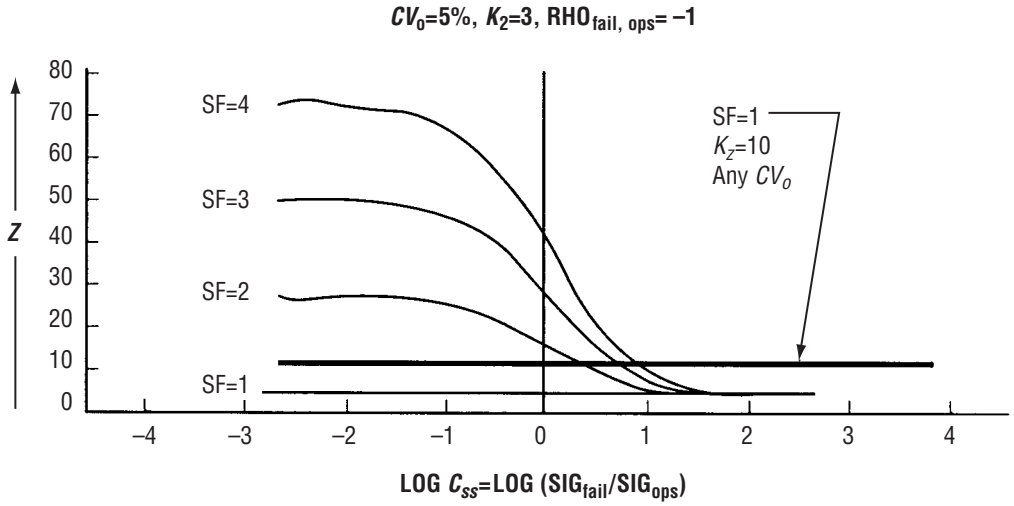


Figure 37. SF effects.

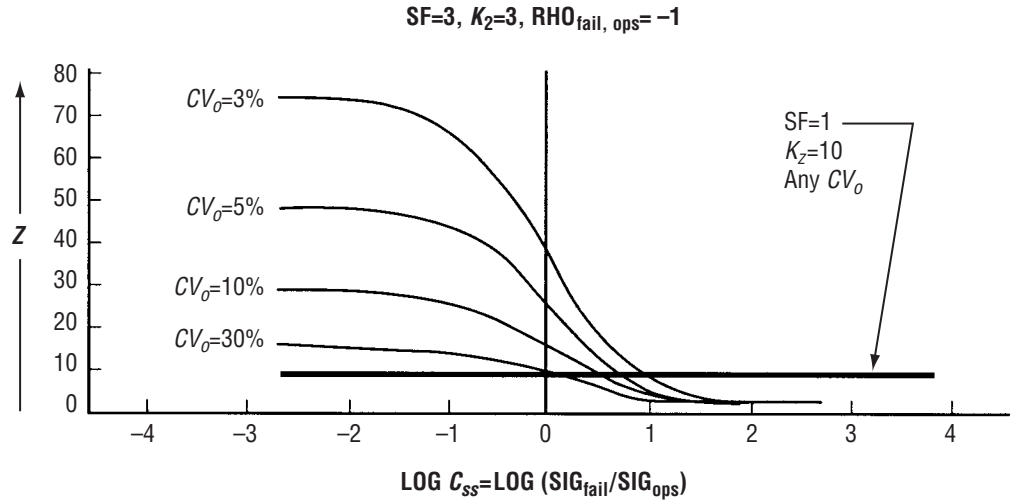


Figure 38. CV_o effects.

For example, an SF of 3 with K_z = 3 for both operational and failure loads, could result in Z between ~3σ and ~50 σ. Although the failure rate corresponding to Z = 50 is not readily available, Z = 8 delivers a failure rate of ≈1 out of 1,500 trillion for a normal distribution.

If the same SF and K_z values are used for a large complex system, wherein CV_o and C_{ss} vary widely from one failure mode to another, some failure modes would be grossly over designed (very large Z) and others marginal designed (Z slightly greater than K_z). If dry weight is a resource spent to avoid hardware failure, then the traditional SF approach tends to misallocate resources. The larger the SF, the larger the dry weight misallocation. In such a case, a few marginal modes would decide the system failure rate and overdesigned failure modes could cost appreciable payload. The use of the traditional SF approach practically guarantees problems in development. If the traditional approach resulted in an acceptable hardware failure rate, cost, and performance, despite the misallocation of dry weight resources, then failure rate, cost, and/or performance can be improved by simply reducing misallocation. In other words, if SF's are wrong, and acceptable hardware is still developed, then using a less wrong method will result in better hardware. Perfection is not required nor is it possible. Any new method must be significantly better or the resulting improvement will not be worth the cost and pain of making the transition.

Misallocation can be almost eliminated, as measured by Z, due to variations in CV_o and C_{ss}, by using SF=1 and by using the same K_z value for both the operational load and the failure load. It can be shown that, under these circumstances, the Z will always be $\geq K_z$. Figure 39 shows that all misallocation cannot be totally eliminated. For example, at K_z=4, C_{ss}=1, and $RHO_{fail, ops}$ =0, Z=1.414×4=5.657. This is the maximum misallocation (0.414 K_z). If C_{ss} differs very much from 1.00 and/or $RHO_{fail, ops}$ is negative, misallocation will be much less. It is doubtful if very many failure modes approximate the criteria of C_{ss}=l and $RHO_{fail, ops}$=0. Regardless, this misallocation is a lot less than would be experienced by using SF's and unequal K_z's.

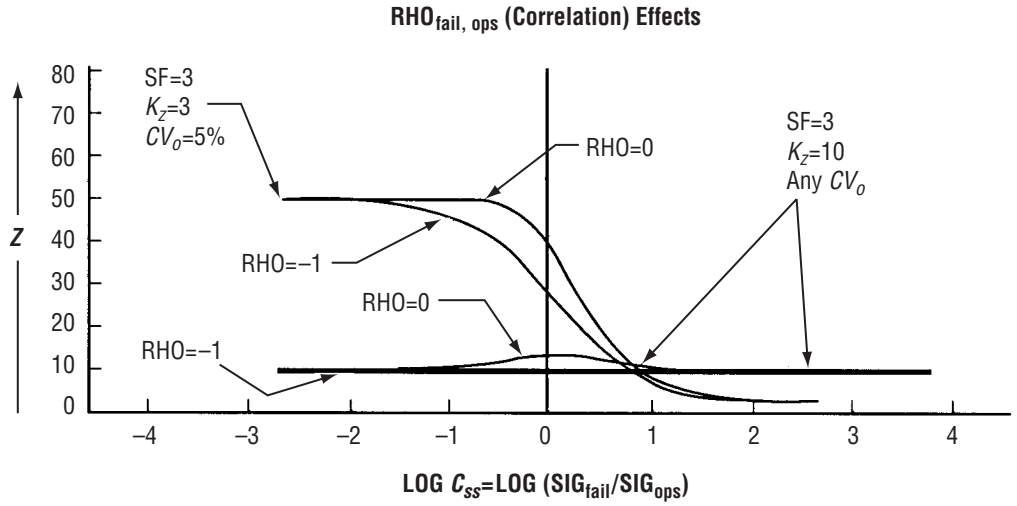


Figure 39. Correlation effects.

The use of SF=1 and $K_z = K_z$ to reduce misallocation has some interesting and useful implications. If, for example, Z=6 is required, set both K_z's=6, the result will be $Z \geq 6$, despite CV_o and C_{ss} values. This permits direct allocation of resources, rather than allowing a haphazard allocation due to variations in SF, CV_o, and C_{ss}. Previous designs used deterministic stress analyses based on 3 σ or worst-case loads, "A" basis material properties, worst-case geometry (maximum and minimum specification limits), and some SF. The use of 6-σ loads, 6-σ material properties and SF=l would have no impact on the method of analysis; it simply changes the input parameters. If 6 σ is not appropriate, any required K_z may be specified, as long as the K_z's are equal.

If the variability of structure strength due to variation in structure geometry is small in comparison to the variation due to strength of materials (as is usually the case), then for all practical purposes, there will be no significant difference between the $Z \geq 6$ promised by this technique and the exact Z for worst-case conditions of CV_o, C_{ss}, and $RHO_{fail, ops}$. In many cases, the conservatism of this approach and using worst-case geometry will accommodate the variability of load capability due to geometry variations.

There are significant administrative advantages to this method. For example, if one contractor is responsible for the load that a structure sees and another is responsible for the load that the structure will carry, set a design limit at, say, 1,000 lb and tell the load contractor that the load has to be 6 σ below 1,000 lb and the structure contractor that the structure strength must be 6 σ above 1,000 lb. This would result in $Z \geq 6$ across this interface without much coordination between these two contractors. This advantage also applies to different departments and disciplines within the same organization.

Being able to treat the operational load and the failure load independently has other advantages. If $Z \geq 3.72$ (99.99 percent for a normal distribution) is required at a 90-percent confidence level for a specific failure mode, ≈100 fairly inexpensive tests to measure loads and 5 expensive test-to-failures to measure strength would be run. If the average load was 4.1275 σ below the 1,000-lb design limit and the average failure was 7.3210 σ above the design limit, the result would be 90-percent confidence that the true $Z \geq 3.72$.

If the relation between the Z and K_z is also valid for probabilities, a failure rate is set ≤1 out of 10,000 for a specific failure mode by setting the load such that it has <1 chance out of 10,000 of exceeding the design limit (say, 1,000 lb), and the structure strength is set such that it has <1 chance out of 10,000 of falling below the design limit. This greatly simplifies the problem of the load distribution and the failure distribution being different. Each contractor looks up the appropriate K_z factor for the distribution and designs hardware accordingly.

This approach also greatly simplifies the extreme value problem. For example, given a turbine with 300 blades and a useful life of 20 flights, then (using tables of normal extreme values) set the turbine blade design limit at 4.42 σ (20 flights) of the test-to-test turbine blade load above the engine specification limit and set the average blade strength at 4.97 σ (300 blades) above the design limit. Assuming no cumulative damage, the odds of an engine losing any blade in 20 flights will be <1 in 10,000. The assumption of no cumulative damage is not generally realistic for rocket engine turbine blades, but it served to illustrate this point. The assumption that the variation in geometry is accommodated by using worst-case geometry limits is less valid for the extreme value problem, but using the engine specification limit should more than compensate for this deficiency.

A.l.4 Contingency Factor *(E)*

Since the SF has been effectively eliminated (SF=1) as a contingency factor, there is a need for a new contingency factor. It can be shown that by derating the average failure load by 20 percent, results in the desired Z despite a 20-percent error in any one of the basic input parameters, AVG_{fail}, AVG_{ops}, SIG_{fail}, or SIG ops. The SF equation becomes:

$$SF=1=(AVG_{fail}\{1-E\}-K_z\,SIG_{fail})(AVG_{ops}+K_z\,SIG_{ops}) \quad , \tag{4}$$

where

E =the desired or required percentage error for this failure mode to tolerate and still deliver a $Z \geq K_z$.

The use of the E factor works simply because a 20-percent change in AVG_{fail} has more impact on Z than a 20-percent change in any other parameter in the SF equation. The error allowance is more of a true SF in that it delivers protection against, say, a 20-percent error, but sometimes that is more protection than needed and sometimes it is not enough.

Unfortunately, this E factor also permits (not causes) the misallocation of resources. If the average failure load is twice the average operational load, then an E of 20 percent will provide protection against a 40-percent shift in the operational load. If E is 20 percent and the CV_o of a parameter is 1 percent, protection against a 20-σ shift is provided. It may be necessary to accommodate a 20-percent error (or more) in some of the engineering models, but it is hard to believe a 20-σ shift in operational load would escape all the safeguards and end up in a flight vehicle. On the other hand, if CV was 30 percent, an E of 20 percent would provide protection against only two-thirds of a sigma shift. It is doubtful if any of the safeguards would detect such a shift in failure load before a failure occurs. In such a case, a 2-σ shift is more important than the 20-percent error in the engineering model.

To design robust hardware (i.e., with an appropriate *Z),* despite engineering model errors and despite process shifts, allowance must be provided for the worst-case input parameters that the safeguards will permit. As in any other design method or design analysis, reasonable worst-case conditions must be used. The reasonable worst-case logic also applies to any Monte Carlo studies. The design must use the worst set of parameters that can escape the safeguards and must do it so that misallocation is minimized.

A.2 Relationship Between Quality Control and Design

Traditionally, the design engineers and design analysts have based their efforts on the assumption that all parameters are within QC specification limits and the SF takes care of any mismatch between the real world and their assumptions. The current NASA QC system is not required, nor designed, to perform to any specific degree of effectiveness. The effectiveness of any particular procedure is determined by a QC engineer's design and selection of a specific sampling/measurement scheme. Although the risk of an out-specification-parameter escaping rejection by the QC system is decided by this QC engineer's procedure, that risk is seldom calculated and transmitted to the design engineer in a useful form.

A.2.1 Quality Control Background

Historically, aerospace vehicle QC has had some problems. About 10 yr ago, the "fastener" scandal triggered massive inspections and reinspections. Huge numbers of defective and suspect fasteners were found in aerospace inventories. Congress passed laws. The American Society of Mechanical Engineers was asked to help. People were fined and sent to jail. After 10 yr, the problem has not been totally corrected. Occasionally, reports of similar problems surface in the QC ALERT system and in newspapers.

This problem was not unique to NASA. Not only did this event prove the nation's QC system ineffective (at least for fasteners), but it also proved that a high percentage of the fastener industry knew that the QC system was ineffective. (How many people would intentionally ship defective hardware to a customer if they knew they would be caught?) There is nothing to preclude a similar event for any other commodity-type items. This was more of a QC scandal than a fastener scandal.

For at least 30 yr, NASA contracts have invoked MIL–STD–4l4 and MW–ST–105 as standard QC sampling plans. Both plans are designed to protect the seller of a product, not the user of that product. For manned flight, the opposite should be true. The bias in both plans is evident from the following illustrations:

- MIL–ST–105:[49] For an acceptable quality level (AQL) of 0.01 percent and a sample size of five, the seller would be 99-percent "sure" that the lot would be accepted if the true defect rate was 0.01 percent, but the defect rate would have to be 60 percent before the design engineer could be 99-percent "sure" that a lot would be rejected. If the design engineer wants hardware to work 99 percent of the time, it would have to be designed to tolerate a 60-percent defect rate.

- MW–STD–414:[50] For an AQL of 1 percent and a sample size of five, the acceptance *K* factor is 1.53. The 1.53 factor is less than the 2.326 factor one would expect from a normal process that generated a 1-percent defect rate. To be 95-percent "sure" that the defect rate is no more than 1 percent, the buyer would need an acceptance *K* factor of 5.749.

Both plans have many options that vary the user's risk. The design engineer is seldom involved in the selection of these options.

A.2.2 Design Threat

SF's, safety indices, and Monte Carlo models are based on assumptions about averages, standard deviations, and distributions. Many of these assumptions are based on data gathered at some point in time which represented only a "snap shot" (e.g., "A" basis). It seems a bit optimistic and risky to assume that these "snap shots" of a process are going to be valid forever.

No process is perfect. All parameters cannot remain in a state of absolutely perfect statistical control. Despite all efforts, including TQM and Taguchi methods, some out-of-control events will occur if any given process runs long enough. Given enough processes, every vehicle and flight will be endangered by many out-of-control events or conditions. Statistical control limits tend to be, or should be, well inside the QC specification limits; otherwise, there would be little point in having the control limits. An out-of-control event may be a 2-σ process shift that triggers some corrective action for future process output, but if the QC specification is still 3-σ away, there will not be any significant number of rejections. Therefore, the output from that shifted process is delivered to flight hardware. A within-specification, out-of-control condition (little or no QC rejections) may be dangerous because the averages, standard deviations, and distributions assumed for design are not being achieved. If the out-of-control condition causes a significant QC rejection rate, then the averages, standard deviations, and distributions delivered to flight hardware are modified even more.

These out-of-specification and out-of-control events are not just because of random variation in a steady-state process, but are sometimes due to the unexpected results of an "obvious" improvement. Some are due to accidents and mistakes, still others may be due to spasmodic out-of-control events of unknown cause. Sometimes the problem just "goes away" before the cause is found, but corrective action was taken anyway, in hope that something was done right. Given that an out-of-control event occurs and is detected, corrective action may or may not be taken. A process may be statistically out of control, but the degree of out of control may be insufficient to trigger action. The process needs some leeway; otherwise, it might tend to over control. For example, if corrective action was taken every time a data point appeared 1 σ away for the process average, the corrective action might drive the process to a random saw-tooth output. Notice that efforts to control the process can cause a process shift. The tighter the control limits, the more false alarms that will be realized and the more false corrective actions taken. As the control limits are expanded, the odds of missing a true alarm are increased.

Several out-of-control items may be produced and accepted before the out-of-control condition is recognized as being outside the control action limits. Even more may be produced before the cause is determined and the problem fixed, since the first fix attempt does not always work as expected. If the system can stand the increased reject/rework rate, if the expected duration of the problem is short, and if some customers urgently need the output to meet a schedule, the process will probably not be shut down. If overtime is required to make up for the increased reject/rework rate, the "quality" of the output may tend to decline even further. If the process line is shut down, the catch-up effort may also produce lower "quality" items until more normal operations can be resumed.

The customer may be totally unaware of the problem, unless there is a significant schedule impact. If the customer samples the incoming product, he may notice that, while it is not quite the same as previous deliveries, it meets all contract QC requirements. Even if the customer is aware of the out-of-control problem, he has no legal nor technical basis for rejecting the hardware, if it is within contract QC specification limits. Deterministic analyses say that the hardware will work adequately if everything is within specification. Except for some sampling plans, most contracts and specifications do not address averages, standard deviations, and distributions. Usually, the number of data points used in a sampling plan is insufficient to draw any worthwhile conclusions about the distribution of any given lot. Hence, from first occurrence of an undetected process shift through all the potential trauma of detecting the shift, finding the cause, fixing it, verifying the fix, and getting back to normal operations, all the failure modes influenced by this process are at some increased risk. Out-of-control events are seldom considered improvements.

Any design based on the assumption that all parameters are "within control" all the time may be a very fragile design. If SF's are used, the overdesigned failure modes may easily tolerate such conditions. A few of the more marginal failure modes will have problems, but after 100 engines, 10 yr, and 2,000 tests, most of these problems will reveal themselves and can be fixed. If some form of the *Z* method or Monte Carlo method is used to reduce misallocation, then many failure modes will be sensitive to out-of-control events and conditions. All failure modes tend to be fragile. Generally, these out-of-control events will not be revealed unless they cause a significant schedule delay, a very costly QC reject rate, or a hardware failure/anomaly. The "out-of-control" scenario is an indication of the real world problems that must be addressed. If ignored, much of this effort will differ little from an expensive academic exercise.

Hardware must be designed to work adequately, despite the uncertainty about the actual averages, standard deviations, and distributions of parameters. The design criteria must render the hardware largely immune to process shifts, whether known or not.

A.2.3 Safeguards/QualityControl

Safeguards are all those activities done to ensure that a specific flight set of hardware is adequate to launch. It includes all inspections, measurements, proof tests, green runs, hot-fire acceptance tests, launch commit criteria, checkouts, etc. In a broad sense, all of this is a QC function. The fact that the people performing these functions may or may not wear a QC "hat" has nothing to do with it. A mechanic who sticks a micrometer to the workpiece in his lathe and decides to continue turning or to scrap the piece is performing a QC function. Sometimes the QC function is merely to note that some hardware was tested per some requirements, and it did not break. But much of the QC function consists of taking some measurement, comparing the result with some specification limit, and taking an appropriate action. The measurement may be a weather measurement, the diameter of a bolt, or a sophisticated prediction of flight performance. In this case, the flight prediction is the measurement, and the computer program and its inputs are the measurement devices. If the program predicts a flight failure, the flight option would be "rejected" and another one selected.

When it is decided to accept or reject something because of a measurement, one is, in effect, making a prediction that the hardware will, or will not, be adequate to fly. If an engine is committed to flight after it passes hot-fire acceptance tests, it has been predicted adequate to fly. Perfect measurements/predictions of hardware in the real world are nonexistent, since all are in error to some extent. There is no perfect

correlation between the measurement taken and the parameter of interest. Part of the error may be due to inaccuracies in the measurement device, the measurement procedures, and/or the skills of the people making the measurement. It should be noted that this error is not the measurement error determined in the calibration lab. It is measurement error at the point where the measurement is taken for making a decision about the acceptance of hardware.

Part of the error may be due to the lack of a physical correlation between the parameter being measured and the parameter of interest. For example, QC testing may be conducted at room temperature to decide that some material will be adequate at 1,000 °F. In some cases, the error will be very small (e.g., diameter of a bolt). In other cases, the error could be large and/or systematically biased. Because of this prediction error, the safeguard system will sometimes reject something that would have been adequate to fly and sometimes it will accept something that is inadequate to fly. If the prediction error is known and sufficient allowance is provided for it, the hardware will work adequately despite the error. This allowance is called a QC design margin.

A.2.4 QC Design Margin

A QC design margin is the difference between a design allowable and the corresponding QC reject limit. The hardware is designed to function successfully at the design allowable; then the QC reject limit is set such that, for all practical purposes, no hardware ever sees the design allowable. In other words, a QC buffer zone (i.e., QC design margin) has been placed between the design allowable and the real world problems that might endanger the hardware (depicted in fig. 40). If the QC system is very effective for that parameter, the QC prediction error will be very small. Hence, the required buffer zone will be very small. If the QC system is not very effective, performance is sacrificed, because the QC buffer zone will be larger.

If the QC prediction error is exactly zero and the *E* for engineering model error is adequate, no design parameter would ever exceed the QC specification limit. The QC system would cleanly truncate all distributions exactly at the QC specification limit, regardless of that distribution's proximity to the QC specification limit. In other words, no load parameter would ever be greater than the specification limit, and no structure would have a strength less than the specification, so the difference between operational load and failure load would always be positive (shown in fig. 41).

Under such conditions, the hardware could be designed based on worst-case QC specification limits and the hardware would never fail. The hardware reliability would be exactly 100 percent at a 100-percent confidence level, despite all the process shifts that might exist. Of course, a QC prediction error of zero does not exist, but the system can be made to behave as if the prediction error is almost zero by providing an allowance for the prediction error. The larger the allowance, the more the system will behave as if the QC prediction error is zero (depicted in fig. 42).

To accomplish this, for example, design the hardware so it works adequately at a stress level of 100,000 psi and place the QC rejection limit at 100,000 + 3.091 × standard deviation of the prediction error (100,000 + QC design margin). For a prediction error standard deviation of 5,000 psi, the QC limit would be 115,455. For a normal distribution and no systematic bias, there would be only 1 chance in 1,000 that material, just barely inside the QC limit, would have a true strength <100,000 psi. The difference between 100,000 and 115,455 is the QC design margin.

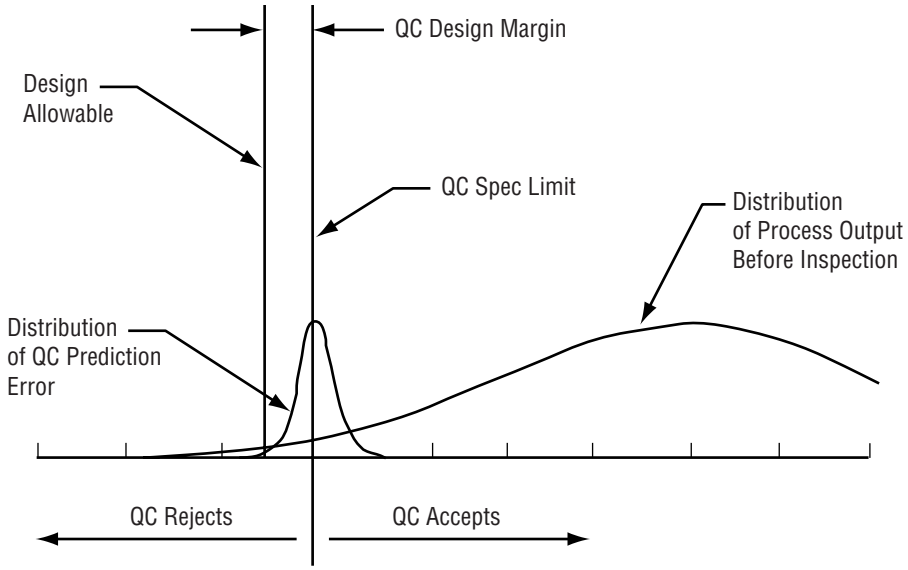


Figure 40. QC design margin.

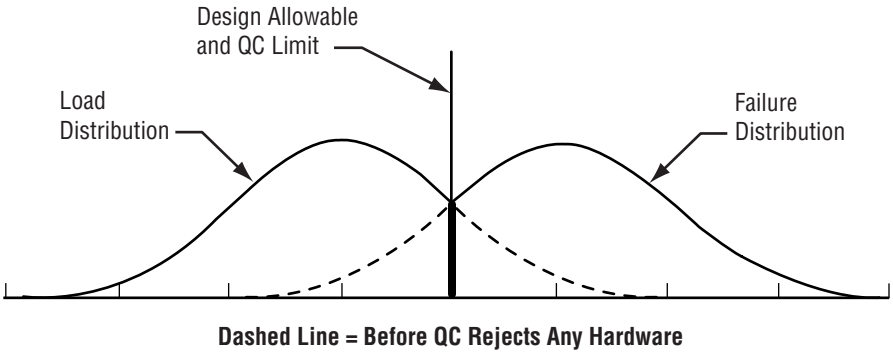


Figure 41. Perfect QC system.

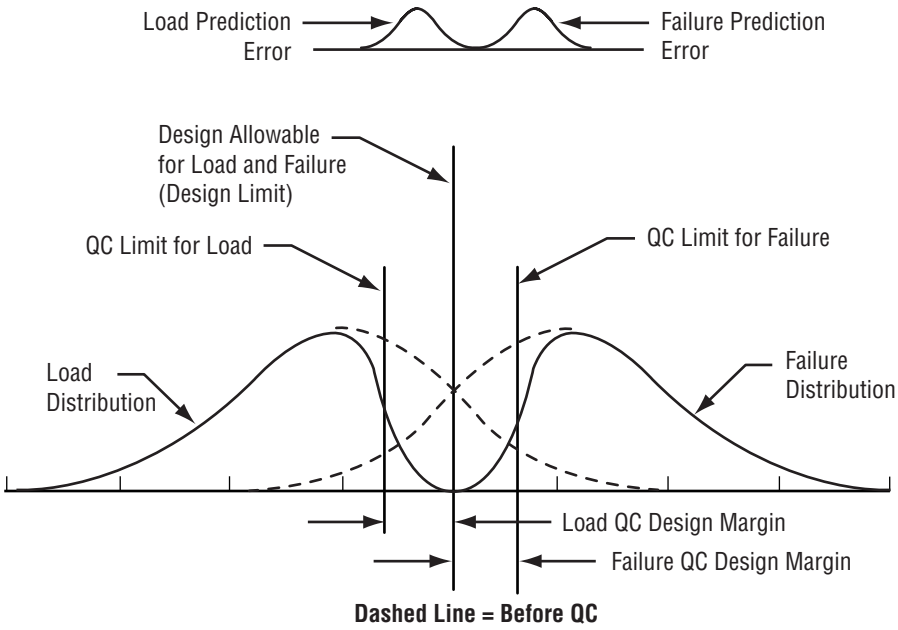


Figure 42. Realistic QC system.

To estimate the QC prediction error for the material strength of a structure, a regression is performed on the failure stress of a structure versus the QC data taken on the same hardware. The QC data may be taken from a witness or tag-end QC test specimen, or maybe just a hardness measurement. The more tests that are conducted, the greater the confidence in the accuracy of the standard error of the prediction and the more one can safely reduce flight weight, but the more the test program costs. For large structures, this will buy the most reduction in total system dry weight and cost the most to determine. For small structures, it hardly seems worth the trouble. It will not cost much, but it will not be worth much to the total system. The value of these tests also depends on the structure's fleet size and the number of flights. A single test program that reduces the weight of several structural elements of the vehicle may be worth the cost, even if the weight reduction per element is fairly small. If the test program can be amortized over many flights, then the test program value is increased.

If a convincing argument can be made that there is some correlation significantly greater than zero and no bias between the QC data and the structure strength, then the standard deviation can be used as the standard error of prediction. If the correlation is truly greater than zero, then the true standard error must be a little less than the standard deviation. The prediction error does not have to be exact, just conservative.

If the lot-to-lot variability is appreciably larger than the within-lot variability, use the within-lot standard deviation as the standard error of prediction. This will buy additional performance. Use a conservative estimate, not the smallest within-lot standard deviation.

If the true correlation is relatively low, but still greater than zero, little would have been gained from the test program anyway. If, however, the true correlation is very high, hence standard deviation of the error (SIG err) is very small, a lot of performance may be sacrificed by not running the test program.

This relation between payload and QC prediction error opens the door to buy additional payload by improving the correlation. Changes in test specimens and test methods may be worth the trouble. If there is doubt as to the existence of any real correlation between the QC data and the structure, there is no control over flight hardware, and the QC system is just wasting money. In this case, corrective action is required.

If the standard deviation (not prediction error) of incoming material properties is greater than zero, then the odds of a true structure strength less the 100,000 psi escaping the QC system would be <1 in 1,000, even if QC is rejecting 50 percent of all incoming material (50-percent rejection rates seldom last very long). If the QC specification limit is, in effect, given (MIL–HDBK–5, MIL specifications, or vendor specifications), then the design allowable would be 3.091 * SIG err below the QC limit.

A typical example of a design scenario for a pressure vessel (no systematic bias nor model error allowance *E)* using this methodology would be as follows:

- Determine the maximum average pressure required.

- Set the QC specification limits on that pressure such that the QC rejection rate will not be too high. If the pressure is outside the QC specification limit for that pressure, the engine would be reworked/modified until it falls within the QC limits. Be very generous with the QC specification limit for engine operational parameters. If there are 100 independent load parameters, a one-sided specification limit of 3.091 σ for each would cause ≈10 percent of the engines to be reworked/modified after the initial acceptance test, although there is nothing wrong. The 10-percent rework rate is simply the result of the normal random variation of these processes. It might be wise to set some control limits at 3.09 σ and put the QC rejection limit at 3.72 σ. This would cut the normal rework rate to ≈1 percent and provide a chance to take some corrective action before rework becomes necessary.

- Compute a design limit of maximum pressure = QC limit + 3.091 * SIG err (maximum QC limit + QC design margin). This is the design load for the structure design allowables.

- Select a set of worst-case parameters (e.g., minimum strength, maximum diameter, minimum wall thickness) to design the vessel so it just barely survives the design limit pressure (i.e., SF= 1).

- Set QC limits on each of these pressure vessel parameters at the design allowable ±3.091 * SIG err. For strength and wall thickness, it should be the design allowable +3.091 * SIG err. For diameter, the QC limit should be the design allowable –3.091 * SIG err, where SIG err for strength, wall thickness, and diameter would all be different. The SIG err for strength could be quite large. The SIG err for wall thickness and diameter would tend to be very small, compared with the one for strength.

Note that this illustration is for a one-sided failure mode. In such a case, the QC design margin for the maximum wall thickness and the minimum diameter could be very small. A pressure vessel, which weighs a little too much or holds a little less fuel, may cost some payload margin, but will not cause a catastrophic failure. Generally, several major components would have to be on the heavy side before any significant payload impact would occur. If, however, the pressure vessel consisted of several components

where a mismatch in diameter or wall thickness could cause one component to induce a significant shear or bending load into another component, a significant QC design margin may be needed for both maximum and minimum conditions. This would apply just to the joint design, given that the joint design provides a good transition to the membrane. If the joint is a small percentage of the pressure vessel dry weight, it would be "zeroed out" by using a large QC design margin or by taking advantage of the correlation between the joint and the membrane failure modes.

For example, the pressure vessel may be designed to work adequately (e.g., just barely escape failure at the design limit) if the material strength was as low as 100,000 psi, a diameter was as much as 36 in., and a wall thickness as little as 0.25 in. However, the hardware would be rejected for any material <120,000 psi, any diameter >35.75 in., or any wall thickness <0.26 in. The differences between design allowables and the QC rejection limit being the QC design margin for each parameter. Notice that the QC margins on the geometry parameters contribute to the effective QC design margin for material strength when material strength is the only parameter in trouble.

Under these conditions and assuming normal distributions, the failure rate for this mode would be <1 in 1,000, even if both the pressure load and any one parameter in the stress equation were experiencing a 50-percent QC rejection rate simultaneously. For any given failure mode, the odds of both the operational load (pressure) and some parameter in the stress equation experiencing a 50-percent QC rejection rate simultaneously would tend to be quite low. But if many flights of a complex system with many such modes were investigated, several modes wherein this worst-case condition is approximated may be found. It is these modes that will be the primary sources of system failure.

The hardware could be designed for a failure rate of <1 in 1,000 when all four parameters (three structure parameters and the load parameter) see a 50-percent rejection rate, although this may be a bit extreme. Under such conditions, the hardware would have to survive a 93.75-percent QC rejection rate before being installed into a flight vehicle. The more complex the structure, the more likely that a "bad" set of hardware will be rejected.

The use of a QC design margin permits the use of deterministic engineering equations and models to design a failure mode to a specified failure rate (e.g., 1 in 10,000 when the design load is perceived to be at the QC limit and no more than one parameter on the structure side of the equation is experiencing a 50-percent rejection rate). The fact that each design parameter is addressed individually and provided the protection according to its needs, reduces misallocation of resources. The fact that each design parameter is addressed individually means that the design engineer only needs to know the statistical properties of the QC prediction error for one parameter at a time and does not have to run a Monte Carlo program to put all the distributions together.

Being able to address one parameter at a time allows the use of simple, general purpose tables, equations, and/or desktop computer programs.

The preceding design scenario for a simple pressure vessel with no systematic bias and no allowance for modeling error was given to illustrate the basic concept. Referring back to figure 42, and considering systematic bias and modeling errors, the general design equations for the QC design margin become:

Design limit = QC limit of maximum load + (AVG load err + SIG load err), (5)

where

Design limit = the load that the structure design allowables must accommodate at SF=1.

AVG load err = the average systematic bias in operational load prediction based on standard QC inputs; if no systematic bias exists, it is zero.

SIG load err = the standard deviation of the operational load prediction based on standard QC inputs; for linear least-square regression, this would be the standard error.

The QC limit for each of the structure allowables are defined as:

MAX QC Limit = (design allowable + AVG err + K_z SIG err)(l+E) (6)

MIN QC Limit = design allowable (1–E) – AVG err – K_z SIG err , (7)

where

MAX QC Limit = the QC limit to protect the lower limit on structure parameters, such as material strength, pressure vessel diameter, and wall thickness.

MIN QC Limit = the QC limit to protect the upper limit on structure parameters, such as material strength, pressure vessel diameter, and wall thickness.

E = the percentage error tolerance desired or needed for this failure mode to tolerate and still deliver $Z \geq K_z$. This judgment factor is now mostly for engineering model error and/or some protection against uncontrollable hardware misuse.

AVG err = the average systematic bias in parameter prediction, based on standard QC inputs. If no systematic bias exists, it is zero.

SIG err = the standard deviation of the structure parameter prediction based on standard QC inputs. For linear least-square regression, this would be the standard error.

K_z = a baseline K factor previously described in section A.1.3.

The AVG err term is a good place to exercise some engineering judgment. If the QC process does not reflect residual stresses in the hardware, it would be better to add an allowance for those stresses to whatever systematic bias may already exist for this parameter for the allowable strength, rather than try to cover it in the E factor. If placed in the E factor, it would penalize all structure parameters. If the QC tests are conducted at room temperature and the hardware operates at 1,000°F, the systematic bias can be quite large, and the prediction error larger than for room-temperature operation. This is also the place to provide an allowance for the worst crack, void, inclusions, and other flaws that might escape the QC system.

If the design requirement is given as $Z=6$, all's K_z's are set to 6. If the design requirement is given in terms of failure rate, the K_z can be found that corresponds to that failure rate for each parameter's prediction error distribution. For example, for a failure rate of 1 in 1 million and a prediction error distribution of a Weibull distribution with a shape factor of 10, the K_z would be 6.117. If the parameter prediction error distribution is normal, a K_z=4.76 would be used. If the prediction error distribution is a Weibull with a shape factor of 2, a K_z = 1.911 would be used. Note that the K_z's are applied to the standard deviation of the prediction error, not to the standard deviation of the parameter in question.

The QC design margin approach has some useful properties. There is very little misallocation of dry weight resources. The method retains the same deterministic models and equations used in the past. Changes are only made to the inputs. Design allowables are used instead of QC specification limits. Variation in the geometry of the structure is of no concern, since it is taken into account. The contingency factor, E (a true SF), is now used to address only the engineering model error, not errors due to model input parameters differing from assumptions. The design limit will bridge the interface between contractors and engineering disciplines. The engineer can address the statistical properties of each parameter and the prediction error for each parameter individually. The maximum failure rate limit is driven by the prediction error. The QC rejection rate is driven by the properties of the parameter process. The failure limits on extreme values can be addressed by using only the statistical properties of the prediction error.

Use of the QC design margin makes the hardware almost immune to out-of-control conditions and makes the properties of the QC system a design parameter. The design specifications and drawings will not only give the geometric parameters with the usual QC tolerance limits (e.g., ± 0.010), but will also require that the standard deviation of the QC prediction error for each of those parameters be no greater than some specified value.

For a sampling plan, the designer may specify that lot acceptance be based on no less than four samples, a sample acceptance K factor of 1.45, and a within-lot standard deviation no greater than some specified value. A general reference to MIL–STD–414, or any other current sampling plan specification, will no longer be sufficient.

Any design without adequate and specific provisions for QC prediction error is an incomplete and fragile design.

A.2.5 Quality Control Design Margin, Organizational Impacts

Use of a QC design margin will change the way business is conducted. There may be much negotiation between designers and QC engineers as they trade QC cost versus performance and failure rate. It may be a new experience for both. QC engineers will be more involved in the mainstream of designing and developing hardware, since much of the QC system will be directly connected to the cost, performance, and failure of the flight hardware. The QC engineers will, in effect, have some design responsibility. Some QC activities have enjoyed a rather vague and anonymous relationship with hardware performance and failure rate. Once the QC system is more directly connected to all performance parameters and failure modes, the system will become more effective. Many QC procedures will have to justify themselves in terms of the tradeoff between cost, flight hardware performance, and failure rate.

There is no well-established infrastructure, customs, practices, nor traditions associated with this new design criteria. Most of the pieces are in place to various degrees, but never before have they been assembled in this fashion. This approach not only permits better engineering, it requires better engineering. One cannot count on the traditional SF to cover mistakes and assumptions. It is undesirable to count on a 10-yr test-fail-fix program of 2,000 + tests to weed out those problems that SF's and the existing QC system do not cover. A 1,000-engine and 20,000-test verification program is not feasible.

In addition to an educational effort on the QC design margin concept, assurances must be provided to take advantage of the "lessons learned" from prior programs. Also, lessons learned from this program on an "as-you-go" basis should be collected and distributed.

A.2.6 Quality Control Design Margin and Testing

When the QC design margin is the design criteria, the primary purpose of all development testing is to measure QC prediction error. Most of the usual development data will incidentally be available. From a failure rate point of view, there is no need for the usual MW–HDBK–5 "A" basis testing. It may be desirable to do some very simple "A" basis type testing, just to be sure that the QC rejection rate will not be too high. If a military or vendor specification provides a limit or an allowable, "A" basis testing is not required. These limits can be used as the QC specification limit.

The "QC basis" is defined as a test program designed to determine the prediction error between some standard operations phase preflight measurement and a flight parameter. Only QC basis is required to design and verify hardware, from a failure rate standpoint. It is suspected that sufficient "A" basis data for estimating QC rejection rates will be incidentally available from QC basis tests.

The difference between QC basis tests and tests typical of a traditional development program may be fairly small, but that difference is critical. For example, there has been an ongoing search for a better way to predict engine performance from preflight data. In the past, the resulting prediction error was not part of the design requirements, so there was no strong incentive to improve the prediction accuracy. Some SF requirement may have precluded any design changes, even if the prediction error could be reduced to nearly zero. For the last several years, there has been talk of using "validated" engineering models. If validation consists of comparing engineering predictions with actuals and concluding that the prediction error is small enough, then the difference between traditional methods and QC basis would be as follows:

- The standard deviation of the prediction error can be used as an input to the design criteria QC design margin. From a failure rate standpoint, the size of the error is unimportant (assuming no bias); but from a performance viewpoint, the error size can make a big difference. Again, use of SF's may have precluded performance gains available from using more accurate engineering models.

- Generally, the discussion of the engineering model accuracy excludes QC measurement error. The QC design margin must include an allowance for the QC measurement error. For example, when testing a pressure vessel, a number of specific QC measurements may be taken at each strain gauge location. During the production/operational phase, these same measurement locations and QC measurements may not be used as the basis for hardware QC acceptance or

rejection. The only engineering prediction error of interest is the one derived from the standard, routine QC measurements planned for this phase of the program. In other words, the prediction error between the standard and special test measurements has been added to the almost pure engineering model prediction error derived from the test program. During the pressure vessel test program, a number of different QC measurements should be investigated to determine which would be best to use during production/operations.

Not all prediction error estimates have to be determined directly from the hardware currently in development. Given that an engineering model has been used on prior programs, one could run a regression model on actuals versus predictions and use this to estimate the prediction error in the region of the current design. If the current design is outside the historical database, the prediction error would have to include an allowance for extrapolation error. Since all engineering models are approximations of reality, a third or fourth order effect in the region of the historical database may be a first or second order effect in the region of the current design. Hence, a small number of tests would be required to confirm that the prediction error for the current design is equal to or less than the error based on historical data. If the current design is well within the bounds of the historical database, the need for confirmation is significantly reduced. The primary purpose of confirmation testing within the bounds of the historical database would be to detect the existence of a mistake, rather than confirming the random prediction error. Figure 43 shows a simplistic example of estimating engineering model prediction error from a historical database.

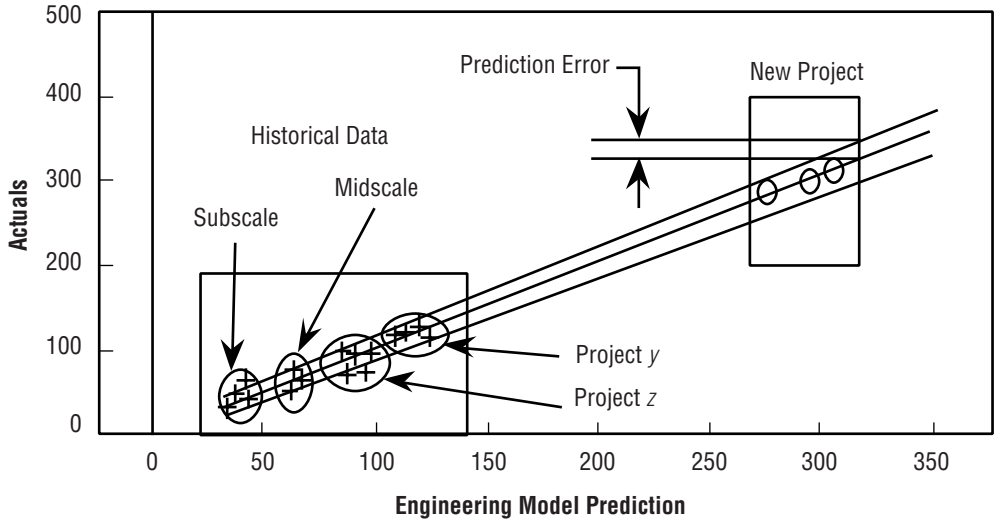


Figure 43. Engineering model prediction error.

As an initial "rule of thumb," it is suggested that the historical database include no less than five different projects with at least six data points per project. This would permit the inclusion of project-to-project variation in the estimated prediction error. If a "fudge" factor is required to remove systematic bias, the same fudge factor will be used for all projects in the database.

Before each engine hot-fire test, a detail prediction of all reasonable parameters will be made. A posttest comparison of actual versus predicted for several engines and tests will be used to estimate prediction error and its behavior over a wide range of possible test conditions and detect any behavior not taken into account by the prediction models. The detection of behavior not taken into account by the prediction models is cause for action.

The predicted and measured parameters will include those that are expected to be highly correlated to determine that they are, in fact, highly correlated and remain so. An unforeseen shift in correlation is cause for action. For parameters that cannot be measured directly, more than one data source will be provided to estimate that parameter via engineering models and significant lack of agreement between the model estimates will be cause for action.

Use of statistical least squares may not be adequate to understand the hardware. To the maximum extent possible, these prediction models are to be basic engineering physics and chemistry models. The prototype/development engines should be so well instrumented that all data possibly needed will be collected. The absolute maximum information from every test should be gathered and maximum use made of it.

Test-to-failure of components, ducts, structures, and pressure vessels will be treated in a similar fashion with many predictions and measurements per test. In this case, however, not only will the exact failure load or pressure be predicted, but the specific failure mode will also be predicted. In some cases, use may be made of subscale items of different sizes to extrapolate the prediction error to a full-scale item, thereby reducing the number of full-scale test items. Sometimes lab tests may be used in combination with engineering/statistical skills to estimate the QC prediction error of a full-scale item at operating conditions.

After an item has been damaged by test-to-failure, it will be dissected to better understand its properties and the correlation between those properties and the nondestructively measured parameters.

A.3 Reliability Verification and Models

This section provides an overview of historical reliability verification approaches and an introduction to the concept of reliability verification through the use of engineering models.

A.3.1 Binomial

The binomial distribution has been the traditional approach to engine reliability validation.[51] This is a simple go or no-go method where some tests are run and a count of the number of tests and failures are made.

Such demonstrations are usually based on the tacit assumption that all tests are of equal value. For example, it is assumed that 100 tests on one engine has the same value as one test each on 100 engines. This assumption is reasonable only if the test-to-test variability is very large in comparison to the engine-to-engine variability. Usually, the real world is just the opposite. Engine-to-engine tends to be much larger than test-to-test, especially for material properties.

It takes too many tests and engines to produce strictly valid and acceptable reliability and confidence numbers. For example, to demonstrate 99.9-percent reliability at a 65-percent confidence level for an expendable engine, a little over 1,000 engines with one test each and no failures would be needed. All engines and tests would need to be identical, and the test duty cycle would need to be the same as the flight duty cycle. A 99.9-percent reliability may not be enough, and 65-percent confidence is definitely not enough. A thousand engines is way too many. For an engine life of 20 flights, with any significant infant mortality and cumulative damage, 20 tests each on 1,000 engines with no failures on any of the 20,000 tests would be needed. Even if true engine reliability was 99.9999999 percent from the very first test, 20,000 tests without a failure would still be required to show 99.9 percent at 65 percent for each test/flight. Of course, there is always the uncertainty about the difference between the flight and ground test conditions.

For obvious reasons, past efforts have been mostly a "going-through-the-motions" activity, generated because there was a requirement for some kind of reliability demonstration.

A.3.2 Reliability Growth

In recent years, there has been a trend toward using reliability growth models to monitor and "demonstrate" reliability (e.g., MIL–HDBK–189).[52, 53] Reliability growth models use more of whatever test data are available, given that enough failures and "growth" are present, than does a binomial model, and therefore "demonstrates" a higher reliability for any given database. In the past, there has been no shortage of failures. The reliability growth model suffers from the same problems as the binomial model. It takes many tests to "show" a high reliability number. If a very reliable engine is designed and developed without going through many "test-fail-fix" cycles, then there will not be enough failures to show growth. In other words, for a successful demonstration using the reliability growth model, the process must be unsuccessful in developing a highly reliable engine in a effective manner.

Almost all reliability growth models are "top-down" (i.e., a least-square fit to history) models. If, during the development of an engine, there are significant changes in the underlying parameters and tacit assumptions, then these "top-down" models can produce strange and unrealistic results.

A.3.3 Engineering Model Verification

One approach would be to build a reasonable worst-case system model consisting of ≈30 design failure modes that are selected based on good rationale (e.g., major dry weight drivers). These failure modes must be detail design failure modes that the design engineer must accommodate, not the "black box" failure modes that are typical of some reliability models. A "black box" failure mode may be defined as the failure of a device to perform some function in a specified manner. There may be a number of different reasons why a device may not perform. The design engineer must identify and address each possible cause. In the case of a large pressure vessel, the "cause" might be rupture, and the major dry

weight driver might be the membrane wall thickness for given material characteristics, worst-case geometry, and load properties.

Next, the model can be used to derive design allowables for those design failure modes, such that the 30-mode system failure rate is acceptable. Design allowables for all other design failure modes are derived such that these other failure modes are "zeroed out." The failure rate due to all the minor modes is so low in comparison to the 30 major modes that the system behaves as if only 30 modes exist. Only 25 of the 30 modes are for specifically identified design failure modes; the other 5 are reserved for design failure modes that may be identified during development.

The model can be fine-tuned by subdividing the system into various subcategories. For example, the major design failure modes may be defined as those design failure modes that drive 80 percent of total dry weight; secondary design failure modes may be defined as those that drive everything between 80 and 90 percent; all others may be defined as minor design failure modes. The system model might consist of 32 failure modes, 25 of which are the top 80 percent. The other two modes would be all secondary and minor modes lumped together.

The more fine-tuned the model, the more optimized the allocation of dry weight resources (e.g., system performance increases as system failure rate is held constant). This model may be fine-tuned to whatever extent desired, but it is likely that the point of diminishing returns will be reached very rapidly. Much of what is gained by using a highly "tuned" model instead of a simple model will evaporate in the transition from preliminary design to a final design. Further, there is no point in using an optimized model (fine-tuned or not) for a subsystem that is a small percentage of total vehicle dry weight. For this to make sense, the total vehicle must be optimized. In such a case, the engine may be a small part of the total vehicle dry weight. Therefore, most or all of its design failure modes may fall into the category of design failure modes to be "zeroed out."

In addition to the weight-based model, other scenarios must be considered. Many design failure modes are not one-sided, but are a "rock-and-a-hard-place" mechanism which cannot be addressed by a simple weight versus failure rate tradeoff. For example, there are design failure modes where the failure rate increases when the metal thickness is too large or too small. The metal thickness has to be 'just right" to minimize failure rate. This is especially true in hardware subjected to environmental extremes. Such extremes generate many of these competitive scenarios. These specific design failure modes are excluded from the dry weight versus failure rate tradeoffs. The failure rate for these specific design failure modes must be consistent with the failure rate requirement of the subsystem within which it resides.

The model must consider failure cost. Some failures may just shut an engine down. Other modes may damage adjacent engines. Those design failure modes that tend to be catastrophic are clearly worth more dry weight investment than those design failure modes that do not tend to be catastrophic. A catastrophic failure could cost the loss of the crew, vehicle, payload, launch pad, several years in delays, additional insurance premiums, and reduced public/customer support.

To further expand the concept, the tradeoff between catastrophic and more benign failures (e.g., mission loss, but safe return) as a function of the relative cost of each event could be explored. Once this tradeoff is understood over a reasonable range, the relation between system failure rate and the average cost per pound of payload could be explored. In this case, cost includes the cost of failure.

The hardware should be built and tested to verify that the design criteria have been met. If successful, the program can conclude that the system reliability is what the model predicts. If the design criteria are not met, the hardware should be redesigned. For example, if the design criterion is a Z of 4 σ and the test data indicates only 3.5 σ, the hardware must be redesigned to meet the 4-σ requirement and then verified. Not having to wait for a failure to trigger a redesign should hasten the transition from the higher failure rate of the initial prototype engine to a low failure rate of a developed engine.

If testing reveals a design failure mode with a Z of 7 σ when 4 σ is required and that mode represents a high percentage of total dry weight, a hardware redesign to 4 σ should be considered. Under these circumstances, redesign is only done if all other design failure modes look good, and the performance gain is deemed worth the cost. Haste to cut the Z may not be warranted since the extra 3 σ may be needed later.

Since the verification of design criteria will consist of measuring averages, standard deviations, and engineering/QC prediction errors, a large number of tests are not required. Verification by variables data requires less tests than verification by the binomial distribution attribute method, but it also requires more skill. For a very simplistic single variable data case, adequate confidence in the estimate is reached in $\approx$30 data points, regardless of the failure rate requirements. For this case, additional data after 30 points do not buy much additional information other than a minimal amount of statistical confidence. After 30 data points, it will be known if the design is adequate. Thus, 20,000 tests are not needed to find out. In the more general case, engineering model verification is more complicated and requires more than 30 data points, but the number of data points required is <20,000. Verification of an engineering model via variables data analysis will require skilled, dedicated personnel. In addition, this model verification approach requires extensive test-to-failure data. Some tests will be expensive. If the environment can be adequately simulated, many of these tests could be conducted at low levels of assembly to minimize cost. However, if successful, the need for a 10^{+}-yr program with several thousand tests will be greatly reduced.

To the best of the authors' knowledge, the model verification approach has never been used for a reliability "demonstration." It is more of a statistically based engineering verification than a statistical verification.

Most of this discussion addresses structural failure rates and some tradeoffs that might be useful for structures. A similar approach can be applied to thermal insulation. Also, a similar approach can be applied to electrical functions, but dry weight probably would not be the best tradeoff parameter. The malfunction warning system where catastrophic failure is traded against mission abort may impact dry weight. The reliability of software code has been specifically excluded from this discussion.

APPENDIX B—Design Reliability Strategy (Conceptual to Detailed Phases)

This appendix provides a more detailed description of the activities discussed in section 4 and outlined in figures 4–6. Included in this are activities appropriate to conceptual, preliminary, and detailed design. The paragraphs in this appendix map to the blocks in the figures by number, preceded by a B for the appendix designator. For example, section B.1.5 correlates to block B.1.5 of figure 4.

B.1 Conceptual Design Phase Activities

This section provides a top-level description of the primary activities applicable to the conceptual phase of the design and development process. The interrelationship of these activities are depicted in figure 4 and discussed in sections B.1.1 through B.1.21.

B.1.1 Customer Requirements

All operability requirements, including main propulsion system reliability, should be specified by the customer at the outset of any program. In the early conceptual design phase, the reliabilities of the overall launch system are generally specified as goals. These goals should include "mission success, "vehicle survival," and "crew survival" reliabilities. For reusable, fast turnaround, high launch rate launch systems "launch on time" should also be specified. The reliability goals should be stated as point probability estimates with the desired level of confidence (e.g., probability of crew survival of 0.999 @ 90-percent confidence). These goals help to define the overall reliability program and its impact on the entire launch system program. High numerical reliability requirements, which are common in the aerospace industry, have significant impact on the design analysis and testing needed to demonstrate these goals.

B.1.2 Program Plan

The program plan defines "how to meet the requirements." This includes mission and vehicle configuration, operations concept, test philosophy, schedules, resource definitions, "who-does-what-to-who," costs, and other programmatic issues and requirements. The reliability part of this plan should address the issues of how reliability will be obtained and how it will be demonstrated.

B.1.3 Conceptual Design Requirements and Ground Rules

Given the program requirements and goals, the vehicle and propulsion systems requirements and ground rules can be derived and established. These include thrust and engine cycle requirements, numbers of engines, payload capacity, reliability, cost, weight, and turnaround time required to meet the launch system programmatic requirements.

B.1.4 Design Allocations

Given the design requirements and ground rules, downward allocations of reliability are made. Reliability allocations should be made at the system, subsystem, and component levels of assembly. These allocations are usually made based on simple "AND" logic using historical reliability information and engineering judgment. "AND" logic is the multiplicative product of the individual reliabilities. Several other techniques are available for the allocation process, but are not addressed herein. This is the most simplistic method of allocation and is adequate for this phase of activity. Historical reliability information is used as a basis for first cuts at modifying the allocation. The original allocation numbers are modified using engineering judgment to account for differences between the historical hardware and the concept hardware. This usually involves consideration of new design tools and philosophies, better quality assurance (QA), improvements in materials, and technological advances. It must be remembered that for each reduction in reliability, an equal increase in reliability in another area is required to maintain the same system reliability.

B.1.5 Conceptual Design Tradeoff Studies

The potential exists for numerous conceptual designs at each hardware level. Multiple trade studies will be conducted using reliability, cost, weight, and other operability and performance parameters in an effort to optimize the design and meet all the goals and requirements. These elements are addressed further in sections B.1.6 through B.l.19.

B.1.6 Historical Cost Database

Historical cost data on components, subsystems, and systems should be developed. Primary cost elements should include design development, test and evaluation (DDT&E), production, operations, and program shutdown. Ideally, each of these primary cost elements should be further developed into higher fidelity categories. DDT&E costs should be segregated into design, test, development hardware, and technology development. Unit production costs should be captured such that learning curve effects can be properly characterized. Operations costs should be segregated into prelaunch, launch, and postflight. Program shutdown costs can stand alone. Approximate costs of unreliability should also be developed to support risk assessments.

B.1.7 Life-Cycle Cost Model

Using historical cost and operations data, combined with engineering judgment and the requirements of the program plan, a life-cycle cost model should be developed for each conceptual design. This model should include the same cost elements defined in the historical cost database efforts of section B.1.6. Significant engineering judgment will be necessary in defining differences in the new program and historical programs. These differences should include considerations of design philosophy, design tools, materials and manufacturing advancement, operational efficiencies, level of QA, and cost of unreliability. It is desirable to use process flow modeling techniques for accurate model predictions, although parametric analysis may be used when sufficient data are available. The life-cycle model should provide pessimistic, optimistic, and expected costs.

B.1.8 Cost Estimates and Predictions

The end results of sections B.1.6 and B.1.7 will be pessimistic, optimistic, and expected estimates and predictions of the component, subsystem, and system costs for each program phase and for each concept design to be used in the conceptual design trade studies.

B.1.9 Engine Performance Model

Given concept guidelines including types of propellants, engine cycle type, thrust class, thrust/weight, and Isp, engine performance studies can be conducted to evaluate concepts. Performance models, such as the engine power balance model, can provide characteristics of the engine operation. This information is used in an overall vehicle performance study to determine typical vehicle performance characteristics including payload to orbit, loads, and heat rates. Predicted operating characteristics and configuration assumptions of the engine concept are key inputs to the engine reliability model.

B.1.10 Vehicle Performance Model

Using engine performance data, a mission model, and appropriate sizes and weights, vehicle trajectory performance data can be generated, executed in a trajectory model, and evaluated to compare concepts. This model executes in a tight feedback loop with a sizing model, iteratively calculating vehicle gross liftoff weight and ascent performance. Early loads and controls analyses provide data supporting early performance, size, and weight estimates. Early engine-out capabilities analysis provides critical insight into off-nominal performance drivers.

B.1.11 Size/Weight Estimates and Predictions

Coupled with a mission model, vehicle configuration studies, an ascent performance model, and other vehicle studies, a database of subsystem weights and mass properties is maintained to support design studies. Many iterations will be required to converge the data to that needed for the next preliminary design step.

B.1.12 Performance Estimates

The end results of the engine performance and vehicle performance model runs will be estimates and predictions of vehicle and propulsion systems' performance including engine Isp and thrust, payload, loads, and heat rates for each concept under study.

B.1.13 Historical Operability Database

This database will include both reliability and maintainability information relevant to the future systems. Past system studies will provide critical operability information including mean times to fail and repair. Failure information should include extensive identification of types and causes of failures and repair times including times to detect, isolate, technician repair time, administrative time, and time for support logistics activities. These data will support both reliability and operations modeling and analysis. Historical reliability data on components, subsystems, and systems should be developed. Primary reliability

elements should include/address mean-time-between-failure (MTB F), mean-time-to-failure (MTTF), mean-time-to-repair (MTTR), time-to-failure distributions, time-to-repair distributions, reliability growth, and infant mortality. It is most advantageous to develop this information to the component failure mode level of fidelity. See section 4.2 for further discussion of reliability growth.

B.1.14 Operations Model

Design operations models are required to support conceptual design studies. Models such as discrete event simulation flows will support timeline and resource requirement studies. Coupled with accurate data such as MTTR information, the model can provide insight into overall timelines, resource usage rates, resource requirements, and resource bottlenecks. Overall performance measures such as hardware availability, process system availability, and launch dependability will also be generated. A key input to any operations analysis is hardware reliability information.

B.1.15 Similarities and Engineering Judgment

Conceptual design operability analyses will be based on operations concepts and operations flows from similarly configured systems. Some hardware may be existing or from very similar systems, but most operations activities will be projected from a historical system. Significant engineering judgment will be necessary in defining differences in the new program and historical programs. These differences should include considerations of operations philosophy, facilities and support equipment definition, material and manufacturing advancement, robustness of systems, and level of QA. However, if good historical data exist from similar systems, laying out proposed operations flows traced to differences from an existing system should provide a reasonably accurate conceptual design comparison. No actual reliability information can be developed for conceptual design hardware. Some hardware which forms a part of the component, subsystem, or system may be existing, well-tested hardware, but care should be taken in considering any differences in the operating environment to which it is subjected. In the majority of cases, the reliability data required must be derived or estimated based upon similarity to historical systems and engineering judgment. Significant engineering judgment will be necessary in defining differences in the new program and historical programs. See section 4.2 for further discussion of reliability growth.

B.1.16 Operations Estimates and Predictions

Using a design operations model and the operations concept document that lays out the maintenance strategy, support equipment, and facility options, operations estimates can be generated that provide insight into operations costs and schedules. Such parameters of interest for operations scheduling include launch dependability and process and hardware availability. Results such as these support trades with hardware reliability, maintenance concepts, and life-cycle costs.

B.1.17 Reliability Database Development

Historical problem reporting and corrective action systems have been inadequate for reliability tracking and trending purposes. Minimal manufacturing, materials, and operational environment (actual and predicted) information has been appropriately databased for historical programs. These deficiencies should be remedied in any new launch vehicle program. As a starting point, the requirements of the STS

problem reporting and corrective action (PRACA)[33] and TRACER[47] database systems should be merged with test, operations, and manufacturing databases to provide a more cohesive environment for data analysis. This database would provide piece-part anomalous conditions with careful documentation of the part history.

B.1.18 Reliability Model

During the conceptual design phase, reliability modeling of the new system will be limited to the component failure mode level of fidelity. Logic modeling employing a top-down approach is the most cost effective and offers considerably higher accuracy than the "parts count" methods normally used at this phase of design. Modeling should be conducted in "failure space" to call specific attention to the failure modes, their effects, and specific mitigators to each failure mode, allowing for direct application of lessons learned based on historical failures.

This modeling technique is in contrast to classical "success space" reliability block diagrams, which do not meet any of these needs. More recent applications and examples of the reliability block diagram method of modeling have begun to include the primary failure modes of components, but only as lists under a higher level block (usually component or subsystem). Since no logic for the propagation or mitigation of the failure mode is represented, accurate quantification of the top event(s) of interest cannot be achieved based on predicted failure mode and mitigation probabilities. Furthermore, the propagation logic from the failure mode to the top event(s) must be fully developed for adequate visibility to the designer, who is attempting to apply countermeasures against these failure modes. Modeling in "failure space" resolves these deficiencies as well as allowing the designer to quantify the effectiveness of the countermeasures.

Due to the lack of design maturity of the propulsion system during this phase of the design (simple schematics), historical information is used to develop failure modes. The primary modes should be identified for all major functional failures. This represents a level of detail such as "valve fails open," "valve fails closed," and "valve fails as is." The logic model should be developed to reflect the propagation of the effects of each of these modes to the top event or events of interest. Mitigating events are then added to the model. At the conceptual design level of maturity, control systems scenarios are generally not well defined, thus control mitigators cannot be modeled. The "work around" to this problem is to include typical "redline safety system works" mitigators based on historical control system philosophies and in coordination with the controls engineers. At this level, multiple logic models will be required to examine the multiple phases of engine system operation. The first stage of each model development can be conducted independently and then merged to reflect system degradation and effects across the multiple operational phases (example, main fuel valve and associated prevalve both fail open during mainstage, prohibiting cutoff of this engine during the shutdown phase). The model(s) are quantified using the data/judgment outlined in sections B.l.13 and B.1.15.

Models at this level of detail and of this type are to be considered very qualitative. Their primary purpose is for comparing differing conceptual designs.

B.1.19 Reliability Estimates and Predictions

The end results of the efforts of sections B.1.18 will be pessimistic, optimistic, and expected estimates and predictions of the component, subsystem, and system reliability measures or each concept design to be used in the conceptual design trade studies (sec. B.1.5).

B.1.20 Conceptual Design Selection

Based on the results of the conceptual design trades studies (sec. B.1.5), one primary and maybe two secondary design concepts are selected. This selection is based on their ability to meet the goals and requirements at an acceptable level of risk while optimizing cost, reliability/operability, and performance. Primary and secondary design selections should have properly documented quantification and modeling justifications to substantiate their selection. These concepts will be carried into the next design phase.

B.1.21 Conceptual Design Performance, Operability, and Cost Predictions

Designs and predictions for all concepts to be carried into the next design phase should be documented with their appropriate justifications and before delivery to the customer. A conceptual design review will be conducted at this point. Some iterations of the concepts may result from this review. This phase will end after the successful completion of the conceptual design review with the customer.

B.2 Preliminary Design Phase Activities

This section provides a top-level description of the primary activities applicable to the preliminary phase of the design and development process. The interrelationship of these activities are depicted in figure 5 and discussed in sections B.2.1 through B.2.23.

B.2.1 Preliminary Design Support Plans

These plans define the necessary testing required to reduce the uncertainty in the databases and engineering judgment used to develop the conceptual design reliability estimates. The primary areas of testing required will be in manufacturing, materials properties, subscale operating environment, and design tool characterization. These are necessary to support a more probabilistically based reliability program and design philosophy.

B.2.2 Preliminary Design Goals and Ground Rules

Using the program requirements, the engine requirements and ground rules can be revised based on the knowledge garnered in the conceptual design phase. The goals and ground rules for thrust and cycle requirements, number of engines, payload capacity, reliability, cost, weight, and turnaround time required to meet the launch system programmatic requirements, as established in the conceptual design phase, are further developed to the subsystem, component, and assembly levels.

B.2.3 Preliminary Design Allocations

Using the design requirements and ground rules as well as the reliability goals, downward allocations of reliability are made to the lower levels of assembly. The refined reliability allocations should be reviewed/made at the component, assembly, and subassembly levels. These allocations are usually made based on the logic modeling developed in the conceptual design phase with the refinements applied. Historical reliability, initial probabilistic design characterization assumptions, and additional engineering judgment are used as a basis for the allocations. These numbers should also consider differences such as new design tools and philosophies, better QA, improvements in materials, and technological advances.

B.2.4 Manufacturing and Materials Processes and Properties List

A list of all the manufacturing and materials processes employed in the fabrication of the hardware will be developed. Existing material properties and process information will be reviewed to identify deficiencies in the characterization of variability. All materials properties and processes that have not been adequately characterized will be listed as requiring further testing.

B.2.5 Manufacturing and Materials Test Plans

Manufacturing and materials processes test plans will be developed to remedy the deficiencies identified by the efforts in section B.2.4. Implementation of this plan will be initiated prior to the preliminary design phase and continue throughout this phase. The objective of these tests is to develop a statistically significant database of the primary drivers of the stress and strength variables. The manufacturing process characterization testing should focus on the primary strength variables of dimensional tolerances and process repeatability. This should also include such concerns as weldment and casting porosity, voids, cracks, contamination, and other such flaws. This flaw characterization should investigate the probability of occurrence and detection, as well as the size, shape, and location. Due to the high reusability of the hardware, the materials testing should focus strongly on the high- and low-cycle fatigue (at many stress, stress concentration, temperature, and other environment effects levels) characteristics and their drivers. The testing should also include the materials properties that are primary drivers of the stress variable (modulus, thermal expansion, etc.). The outputs of the materials characterization testing should include, as a minimum, a family of curves for the mean and sigma values. Ideally, the outputs should be a family of distributions. These test programs should be structured using design of experiment techniques to maximize the information gathered, while minimizing the cost involved in meeting the reliability data requirements. These test plans will require significant input from the reliability data requirements development efforts.

B.2.6 Predicted Operating Environment

Operating environment predictions are made throughout the design process using numerous prediction tools. The operating environment provides the basis for the loads the hardware must be designed to endure, including pressures, temperatures, vibration, maneuver loads, rotordynamic forces, impact, and other static and dynamic loadings. A review of the prediction tools and methods should be undertaken to characterize the uncertainty/accuracy of these analyses. Deficiencies in the ability to properly characterize the operation environment and the analysis tools will result in the need to conduct operating environment testing for the development of statistical distributions of the loads.

B.2.7 Operating Environment Test Plans

Test plans will be developed and implemented for all areas of deficiency in characterization of the operating environment, as identified in section B.2.6. These tests will include subscale and model testing for characterization and validation of predicted flows, pressures, temperatures, vibration, rotordynamic forces, impact, and other static and dynamic loadings, as well as the models used to predict these environmental conditions. The outputs of this testing should include, as a minimum, a family of curves for the mean and sigma values. Ideally, the outputs should be a family of distributions. These test programs should be structured using design of experiment techniques to maximize the information gathered, while minimizing the cost involved in meeting the reliability data requirements. These test plans will require significant input from the reliability data requirements development efforts.

B.2.8 Reliability Data Requirements

As stated in section B.1.17, historical problem reporting and corrective action systems have been inadequate for reliability estimation, tracking, and trending purposes. Minimal manufacturing, materials, and operational environment (actual and predicted) information has been appropriately databased for historical programs. These deficiencies should be remedied in any new launch vehicle program. As a starting point, the requirements of the STS PRACA and TRACER database systems should be merged with test, operations, and manufacturing databases to provide a more cohesive environment for data analysis. To the extent that this reliability methodology advocates a "new" design philosophy/criteria, much effort must be expended in the development of data requirements. The basic requirements have been outlined in sections B.l.17, B.2.5, and B.2.7.

B.2.9 Design of Experiments

Statistical methods of experimental design should be used in the test plans to ensure effective and economical results. It is extremely important to develop the experimental designs in a fashion that maximizes the information obtained without masking multifactor interactions. This can be accomplished by using properly developed fractional factorial experiments. The key to proper testing is a fair understanding (and good assumption) of the primary drivers and their interactions. These will usually be developed from prior experience and limited sensitivity testing. Most experimental design texts include significant discussion and examples of proper application of these methods.

B.2.10 Manufacturing Process, Materials, and Environment Subscale Testing

The test plans outlined in sections B.2.5 and B.2.7 should be implemented in a timely fashion in order to establish appropriate design criteria. The materials industry is well ahead of the "analysis" industry in statistical characterization of their products. Many volumes of material properties data have been developed over the years, but the presentation of the material is generally inappropriate to the development and implementation of the type of design philosophy presented herein. The primary problem is the information that is generally given in 2- or 3-σ minimums, whereas the development of the criteria will require material property distributions. The analysis industry has established standards and benchmarks for validating models deterministically, but to the authors' knowledge, little statistical information or statistical testing standards have been developed. These testing efforts should serve to remedy this deficiency.

B.2.11 Variability Estimates

The results of the above testing efforts (sec. B.2.10) should provide estimates of the statistical distributions of the parameters defined in the test plans (sec. B.2.5 and B.2.7). This will provide the necessary variability information for development of design criteria, thus allowing more realistic reliability predictions.

B.2.12 Reliability Data Collection

In addition to the stress and strength parameter information gathered in the test program, all failure data and anomalous conditions information from the testing should be aggregated and analyzed. Comparisons of predicted and actual behavior should be made. Lessons learned with corrective actions should be collected, databased, and considered in the design criteria and the hardware designs. This information should also be compared with historical design, test, and reliability data for consideration in the development of the design criteria. Historical problem reporting and corrective action systems have been inadequate for reliability tracking and trending purposes. Minimal manufacturing, materials, and operational environment (actual and predicted) information has been appropriately databased for historical programs. These deficiencies should be remedied in any new launch vehicle program. As a starting point, the STS PRACA and TRACER database systems should be merged with test, operations, and manufacturing databases to provide a more cohesive environment for data analysis. The basic requirements have been outlined in sections B.1.17, B.2.5, and B.2.7.

B.2.13 Establish Preliminary Design Criteria

Based on all of the above efforts, appropriate design criteria can be developed and implemented as described in the reliability data requirements section (sec. B.2.12) and detailed in section 4.2.

B.2.14 FMEA/CIL

Using the updated reliability logic model and operations models as guides, a preliminary FMEA should be conducted in a bottom-up fashion to ensure that all credible failure modes are identified. In addition to the description of the failure mode and its effects, the FMEA will generally include additional information such as item function, operational mode, failure mitigators, failure detection method, and suggested methods of failure elimination. The FMEA should also depict a critically rating system (Crit. 1, 1R, 2, and 3). It is imperative that the FMEA cover both operations and processing to identify all failure modes. More detailed information on the development of a FMEA and a failure modes, effects and criticality analysis (FMECA) is contained in MIL–STD–1629A, “Procedures for Performing a FMECA.” Upon completion of the FMEA, a CIL should also be developed. This will identify specific areas that must receive additional attention to minimize risk. This will also assist in the proper allocation of resources. Information gathered from this process should be compared with and incorporated into the reliability logic model (sec. B.2.15).

B.2.15 Reliability Logic Model

The reliability logic model(s) developed during the conceptual design phase should be updated and expanded. For complex systems, these efforts are best conducted using models that represent the component failure modes as top events. These lower level models are then aggregated into a system model as described in section B.2.17. For all high-risk critical items, the model should be expanded to the part failure mode level. Special attention should be given to mitigating events such as control system redlines, designed-in "crack stop" features, bill of material object damage elimination features, and others which could possibly eliminate the propagation of FMEA identified failures. These models will serve to evaluate the FMEA and properly quantify the probability of the previously defined top events of interest. In addition, the model should be updated and expanded based on the latest design information. All functional failures having a top-event contribution of greater than some predetermined threshold should be modeled to lower levels in order to ensure proper characterization of the failure modes and quantification. This will allow early designed-in mitigation of these high-risk areas. All previous quantification of the model was conducted as single-point probabilities. At this point in the design process, it will be appropriate to begin the expansion of the model to incorporate time-to-failure distributions. This information will be vital to the operations and maintenance analysis efforts.

B.2.16 Item/Component Reliability Model

Quantification of the above models at the item functional failure mode will be as described in section 4.2. The above models can then be analyzed for component reliability and subsequently integrated into the systems design reliability model described in section B.2.17.

B.2.17 Systems Design Reliability Model

The reliability logic models developed (sec. B.2.15) will be integrated into a single systems-level model. Special attention should be given to propagation paths between the individual component models. This model will serve to evaluate the FMEA and properly quantify the probability of the previously defined top events of interest. In addition, the model should be updated and expanded based on the latest design information affecting the functional interactions of the components. Although this system model may become quite large, the combination of currently available workstation/desktop computer speed and computationally efficient computer programs such as the FEAS–M will allow for reasonable turnaround times for their analysis. This program is especially suited for the development of numerous individual models that can be analyzed independently or merged together for a complete system analysis.

B.2.18 Operations Estimates

The operations model developed during conceptual design should be updated and expanded. As the operations concept becomes more detailed, the analysis can become more detailed. Better manpower estimates and timelines associated with the design become more plausible. A movement away from point estimates to probabilistic estimates becomes necessary. Such probabilistic estimates serve to support an allocation of resources to other analysis areas. A general movement from concept and trade analysis at a more macro level to an analysis of a specific propulsion system design is the trend.

B.2.19 Manufacturing and Supplier Estimates

In order to validate the earlier cost estimates, vendor quotes are gathered at this stage to include in the cost model. Analyses are needed to forecast cost to actual material, manufacturing, manpower, and overhead costs. Estimates should be received from several competing vendors to support this required level of detail and to validate model estimates. Many key design and programmatic decisions hinge on the accuracy of this model and the supporting data acquired during this step. Early estimates of maintenance requirements and life limits are critical to preliminary design cost estimating.

B.2.20 Preliminary Design Cost Model

Operations and supplier estimates, dependent upon the preliminary design goals and ground rules set for this phase, feed the life-cycle cost model and serve to validate the model and earlier design estimates. Again, unreliability effects must be modeled.

B.2.21 Weight Model

An update of the size, weight, and mass properties database occurs after each phase, and often several times within each phase, to keep up with the level of detailed analysis occurring during that phase. While under configuration control, this model and database support critical analyses in all functional areas.

B.2.22 Performance Model

Additional and more accurate weight and mass properties data support more detailed analyses on an actual design concept. Analyses of loads, controllability, and ascent performance become more detailed to include nominal and off-nominal cases. Reference trajectories serve as nominal reference points for most analyses and are done early in this phase. Off-nominal analyses include engine-out and dispersions analyses.

B.2.23 Preliminary Design and Trades

Multiple trade studies which use reliability, cost, weight, and other operability and performance parameters in an effort to optimize the design and meet all the goals and requirements, will be conducted. Based on the results of the preliminary design trades studies, one primary and maybe two secondary design concepts are selected for each component/subassembly. This selection is based on their ability to meet the goals and requirements at an acceptable level of risk while optimizing cost, reliability/operability, and performance. Primary and secondary design selections should have properly documented quantification and modeling justifications to substantiate their selection. These preliminary designs will be carried into the next phase of the design process. A preliminary design review (PDR) will be conducted at this point. Some iteration on the designs may result from this review. This phase will end after the successful completion of the PDR.

B.3 Detail Design Phase Activities

This section provides a top-level description of the primary activities applicable to the detail phase of the design and development process. The interrelationship of these activities are depicted in figure 6 and discussed in sections B.3.1 through B.3.22.

B.3.1 Detail Design Support Plans

Updates of the previous support plans are conducted to define the testing required to reduce the uncertainty in the databases and engineering judgment used to develop the preliminary design reliability estimates. These plans should have a strong focus on the highest risk, lowest reliability areas of the design.

B.3.2 Detail Design Goals and Ground Rules

Using the updated program requirements, the engine, component, subassembly and part level requirements and ground rules can be developed/updated based on the knowledge garnered in the conceptual and preliminary design phases.

B.3.3 Detail Design Allocations

Using the updated design requirements and ground rules, updated downward allocations of reliability are made to the lower levels of assembly. The refined reliability allocations should be reviewed/made at the subassembly and part levels. These allocations are usually made based on the logic model developed in the conceptual design phase with the preliminary design refinements applied. Initial probabilistic design characterization and additional engineering judgment are used as a basis for the allocations.

B.3.4 Reliability Data Requirements

Based on the results of the previous testing and increasing maturity of the design, updates to the reliability data requirements will be required to support the continuing design efforts. Deficiencies identified in or by previous testing and analysis should be addressed to properly impact current and future testing and design decisions. These requirements should also list the additional information required to conduct higher fidelity reliability analysis of high-risk items which were identified in the previous reliability analysis efforts. Concerns and uncertainties raised by the hardware design function should also be addressed. Reliability efforts during this phase of the design will also focus on life analyses to support cost, maintainability, operability, and performance requirements that rely on these reliability inputs.

B.3.5 Manufacturing and Materials Characterization Test Plans

Deficiencies and required updates as identified in the reliability data requirements (sec. B.3.4) should be addressed in the updating of the manufacturing and materials characterization test plans. The extensiveness of the testing required to support this design criteria concept will require that this type of testing be conducted throughout the design process to ensure accurate and adequate reliability estimates to support the many interrelated disciplines. The required outputs of this testing will be as previously stated in the earlier design phases, but will be updated based on lessons learned and new requirements levied by the maturation of the design.

B.3.6 Operating Environment Characterization Test Plans

Deficiencies and required updates as identified in the reliability data requirements (sec. B.3.4) should be addressed in the updating of the operating environment characterization test plans. The extensiveness of the testing required to support this design criteria concept will require that this type of testing be conducted throughout the design process to ensure accurate and adequate reliability estimates to support the many interrelated disciplines. Operating environment testing during this phase of the design will be increased in scope to include subassembly and component testing. Information and lessons learned from previous testing, as well as concerns of the design functions, should be given appropriate consideration in the development of these plans. The tests will include subsystem and component level testing for characterization and validation of predicted flows, pressures, temperatures, and other loads as stated above, as well as the models used to predict these environmental conditions. As during the preliminary design phase, the outputs of this testing should include, as a minimum, a family of curves for the mean and sigma values. Ideally, the outputs should be a family of distributions. These test programs should be structured using design of experiment techniques to maximize the information gathered while minimizing the cost involved in meeting the reliability data requirements. These test plans will require significant input from the reliability data requirements development efforts.

B.3.7 Design of Experiments

Statistical methods of experimental design should be used in the test plans to ensure effective and economical results. It is extremely important to develop the experimental designs in a fashion which maximizes the information obtained without masking multifactor interactions. This can be accomplished by using properly developed fractional factorial experiments. The key to proper testing is a fair understanding (and good assumptions) of the primary drivers and their interactions. These will usually be developed from prior experience and limited sensitivity testing. Most experimental design texts include significant discussion and examples of proper application of these methods.

B.3.8 Manufacturing and Materials Characterization Testing and Component/Subscale Testing Continued

The test plans outlined in sections B.3.5 and B.3.6 should be implemented in a timely fashion in order to update the design criteria. The manufacturing and materials characterization testing will be a continuation of the previous testing with updates as identified in the test plan. The primary focus of the component and subsystem testing will be to validate engineering models and properly characterize the associated uncertainties. Extensive testing at this level will minimize both cost and risk, while developing a statistically significant database of the primary life drivers.

B.3.9 Variability Estimates

The results of the above testing efforts (sec. B.2.8) should provide updates and additional information for estimates of the statistical distributions of the parameters defined in the test plans. This ongoing testing will provide the additional variability information for development of design criteria, thus allowing updating of the reliability predictions and design criteria.

B.3.10 Reliability Data Collection and Analysis

The reliability data collection and analysis efforts will be continually updated during this phase of the design (see sec. B.2.12). As testing time increases during this design phase, additional efforts will be required in the areas of life driver analyses and prediction of wear-out rates.

B.3.11 Design Criteria Update

Based on all of the above efforts, the design criteria should be updated, as described in section B.3.10, and implemented.

B.3.12 FMEA/CIL

Using the updated design, reliability logic models, and operations models as guides, and any additional failure modes identified by the testing efforts, the FMEA/CIL should be updated. Information gathered from this process should be compared with and incorporated into the reliability logic models based on the previously described criteria (sec. B.3.11).

B.3.13 Detail Reliability Logic Models

The reliability logic models developed in section B.2.15 and B.2.17 will be updated to reflect the current design information. The models will require extensive expansion to investigate the effects of detail/part-level failure modes. Correlation between failure modes, between failure modes and mitigators, and between mitigators should be addressed in these models. Special attention should be given to propagation paths between the individual component models. This model will serve to evaluate the FMEA and properly quantify the probability of the previously defined top events of interest. Integration of the component-level models into a single system-level model is unlikely, due to the resulting size. Current state-of-the-art software and desktop computers are not yet up to the challenge of analyzing models of this size with correlated failures included. The previously developed system model (sec. B.2.17) should be updated and used for systems-level analysis using the outputs of the component models.

B.3.14 Detail Probabilistic Design Analysis

In the majority of parts, the design criteria will be met and reliabilities can be estimated. In some cases, due to competing stress and strength parameters, detail probabilistic analyses will be required. These will be conducted using the same methods as used for the development of the design criteria, but will be specific to the particular parameters for the part or subassembly. The analysis will generally focus on the life drivers of the stress and strength parameters. These models will be used to optimize the design and conduct workaround trades. Many times these models will be used when the design maturity does not allow for extensive redesign efforts or incorporation of extensive mitigation and where a more detailed analysis may provide the necessary information to ensure the reliability goals will be met. The results of the above efforts should provide optimistic, pessimistic, and expected time-to-failure distributions for the part, subassembly, and component levels of detail.

If the part is critical enough, a preliminary stress/strength type of analysis may be completed prior to the PDR. This would provide early design decision support and point to any further analysis or data collection needed.

B.3.15 Predicted Part Reliability/Wear-Out Rates

The outputs of the above analyses (secs. B.3.13 and B.3.14) will be the predicted wear-out rates used to support the operations modeling efforts.

B.3.16 Systems Design Reliability Model

The reliability logic model developed during the preliminary design phase should be updated and expanded based on the latest design information. The majority of the previous quantification of the model was conducted as single-point probabilities. Expansion of the model to incorporate time-to-failure distributions from sections B.3.13 and B.3.14 should be the primary focus of the efforts during this phase. This information will be vital to the operations and maintenance analysis efforts. The results of these efforts should provide optimistic, pessimistic, and expected time-to-failure distributions for the part, subassembly, and component levels of detail. The propagation of these distributions in the logic model will provide optimistic, pessimistic, and expected time-to-failure distributions for the top event(s) of interest.

B.3.17 Sensitivity Analysis

Based on the previous modeling efforts (sec. A.6.16), sensitivity analysis should be conducted. This analysis examines the effects of each of the life drivers on the top event. By varying the life drivers by predetermined quantities, their effect on the top event can be quantified and compared. This analysis should also be conducted on the primary life drivers of the probabilistic design analyses to examine and quantify their effects on the part reliabilities. The results of these efforts provide a guide for the appropriate application of resources.

B.3.18 Spares Requirements

Based on the reliability predictions of the previous efforts (sec. B.3.16), spares requirements and provisioning can be determined. This will be significantly impacted by the operations and maintenance program philosophy for determining line replaceable units, depot maintenance, use as is, return to manufacturing facility, rebuild, and other operations and maintenance parameters for the program. These item requirements are direct input requirements for the operations and life-cycle cost models.

B.3.19 Vehicle Life-Cycle Cost Model

The life-cycle cost model should be updated to reflect modifications in the design. This will include hardware, testing, production, and operations effects that the design maturation process has brought about. All reliability, operations, performance, and other models as described earlier should be updated and their predictions included in the model. Any programmatic changes must also be incorporated.

B.3.20 Weight Model

The weight model and database are updated to reflect the design and analysis detail and for any necessary design changes made during preliminary and detailed design phases. Significant weight changes should be well documented and traceable to specific design decisions.

B.3.21 Performance Models

The performance models of the vehicle and propulsion system should be well on their way to being validated using the performance analyses conducted during this and previous phases. Both nominal and off-nominal performances are critical to analyses. Significant drivers of design decisions have often been identified during analysis of off-nominal performance of aerospace launch vehicles. Again, significant performance changes should be well documented and traceable to design decisions.

B.3.22 Detail Design

The result of this design phase is a single design to be baselined for full-scale development hardware manufacturing. All design, performance, reliability, weight, operations, testing, and other efforts described previously should be appropriately documented to support the design decision. Evidence of how all program goals and requirements have been meet should be provided, or, in the case of a goal or requirement not being met, appropriate management approval should be acquired. A critical design review (CDR) will be conducted at this point. Some iteration on the designs may result from this review. This phase of the design and development program will end after the successful completion of the CDR.

APPENDIX C—MPS Qualitative Analysis Support Data

C.1 X–34 MPS Pneumatic Purge System Design Fault Tolerance Analysis Engineering Support

Evaluation of MPS pneumatic purge system failure scenarios required two types of engineering inputs: (1) The evaluation of pneumatic pressure behavior during nominal and off-nominal operation, and (2) an assessment of pneumatic component reliabilities. Given the operational loads induced by the nominal and off-nominal pneumatic pressure profiles and the associated component reliabilities, the risk of system failure initiation and propagation was assessed. The following example analysis illustrates how these inputs were incorporated into the final risk assessment.

C.2 Interpropellant Seal Purge Supply Analysis

The turbopump of the Fastrac engine used in the X–34 vehicle consists of an integrated package of an RP–l pump, a lox pump, and a hot-gas turbine. Propellants within the RP–1 and lox pumps are separated by an IPS to which the MPS supplies a helium purge. This purge maintains propellant separation by providing a positive pressure in the IPS interseal cavity. If this purge is interrupted while propellants are present in the pumps, the propellants may mix, causing a fire or explosion. On the other hand, if a failure mode leads to an overpressure condition in the IPS purge supply, structural damage may occur in the IPS cavity that could lead to a catastrophic event. Significant engineering input was required to assess the risk and consequences of either an overpressurization or a loss of IPS of the IPS cavity helium supply.

The IPS cavity purge supply overpressure failure scenario was evaluated by first predicting the IPS cavity pressure profile, given the maximum pressure profile at the MPS/engine IPS supply interface due to MPS pneumatic system regulation failure and IPS purge system resistances. This maximum pressure profile assumed the failure of the purge system overpressurization mitigation response. Once this pressure profile was established, structural assessments were performed by turbomachinery design and structural analysis engineers. The resulting damage from the overpressure was evaluated by the turbomachinery design engineers for possible propellant mixing within the turbopump.

The conclusions of the IPS cavity purge supply overpressure failure scenario analysis was that a fail open/fail high of the purge supply line regulator could lead to propellant mixing within the engine turbopump within 1–2 sec. The catastrophic risk associated with this propellant mixing failure scenario was evaluated, as discussed later in this section.

The consequences of a loss of IPS cavity purge supply was evaluated in a similar manner as the purge supply overpressure scenario. First, a pressure profile was established in the IPS cavity in the event that the IPS supply is lost, which incorporated the pressure decay rate due to system resistances. Once this pressure profile was established, the turbomachinery design engineers determined the time between loss of IPS supply at the MPS/engine interface and possible propellant mixing in the IPS cavity. The conclusion of this assessment was that a loss of IPS cavity purge supply could lead to turbopump propellant mixing within 1 sec.

The consequence of propellant mixing in the IPS cavity was determined by a two-part analysis. First, the MSFC Industrial Safety Office determined the maximum explosive yield of mixed propellants in the IPS cavity, given the maximum volume of propellants that could mix. Turbomachinery design engineers and structural engineers then evaluated the consequences of the maximum explosive yield. The conclusion of this analysis was that if turbopump propellant mixing occurred while the X–34 vehicle was still attached to the L–1011 carrier, a catastrophic loss of the carrier could occur. If this turbopump propellant mixing occurred after the carrier had released the X–34 vehicle, any resulting X–34 vehicle explosion would occur after the vehicle is a safe distance from the carrier.

This analysis deemed that:

1. The X–34 IPS purge supply system was required to be two-fault tolerant to a loss of pressure and an overpressure failure scenario while the X–34 vehicle is attached to the carrier.

2. The X–34 IPS purge supply system was not required to be two-fault tolerant to a loss of pressure or an overpressure failure scenario after the X–34 vehicle is released from the carrier.

Design decisions based on these analyses are as follows:

1. The MPS IPS purge supply isolation valve may be controlled by controllers aboard the carrier and may be locked into an open position just before vehicle release. The carrier controllers must be able to close the isolation valve within 1 sec if necessary.

2. One of the two IPS purge supply backup sources required for two-fault tolerance during captive carry may be located aboard the carrier. Once the vehicle is released, this second backup purge source is no longer available.

Figure 44 provides the X–34 MPS failure propagation logic models for the pneumatic purge system, a system of critical importance. The propagation logic for credible X–34 MPS pneumatic purge system failures was modeled to verify this MPS system design compliance to the requirement of two-fault tolerance to catastrophic failure and to identify health monitoring requirements and instrumentation. These propagation logic models were developed with the FEAS–M software tool described in section 5.

The failure propagation logic models in figure 44 evaluates failure modes associated with three phases of X–34 MPS pneumatic purge system operation: captive/carry from taxi to propellant drop, captive/carry from propellant drop to vehicle release, and postvehicle release to the engine-start command. All credible catastrophic failure modes were modeled. A failure mode is deemed catastrophic if the result is a loss of carrier and human life. Some noncatastrophic failure modes were modeled to clarify rational in deeming these failure noncatastrophic. The rationale for deeming other failure modes noncatastrophic may be inferred from the following modeling assumptions.

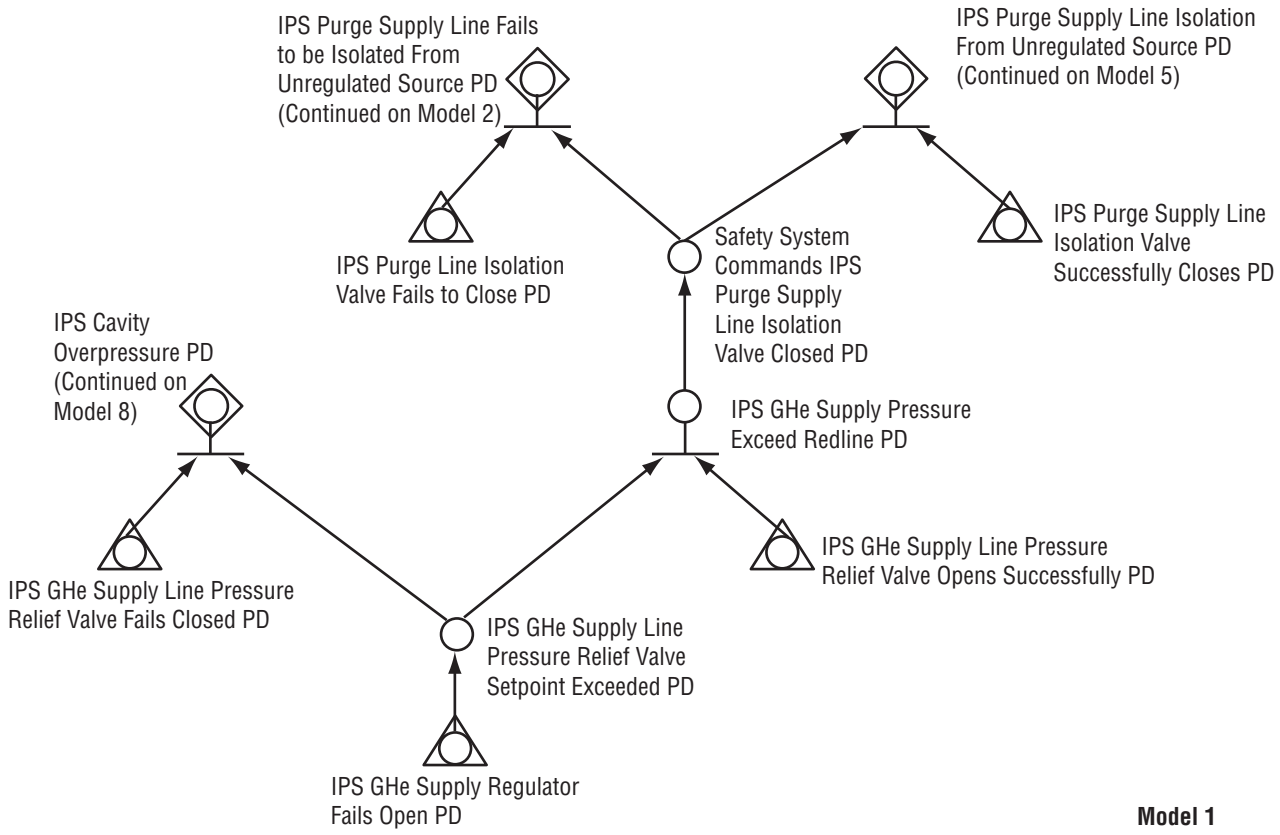


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 1).

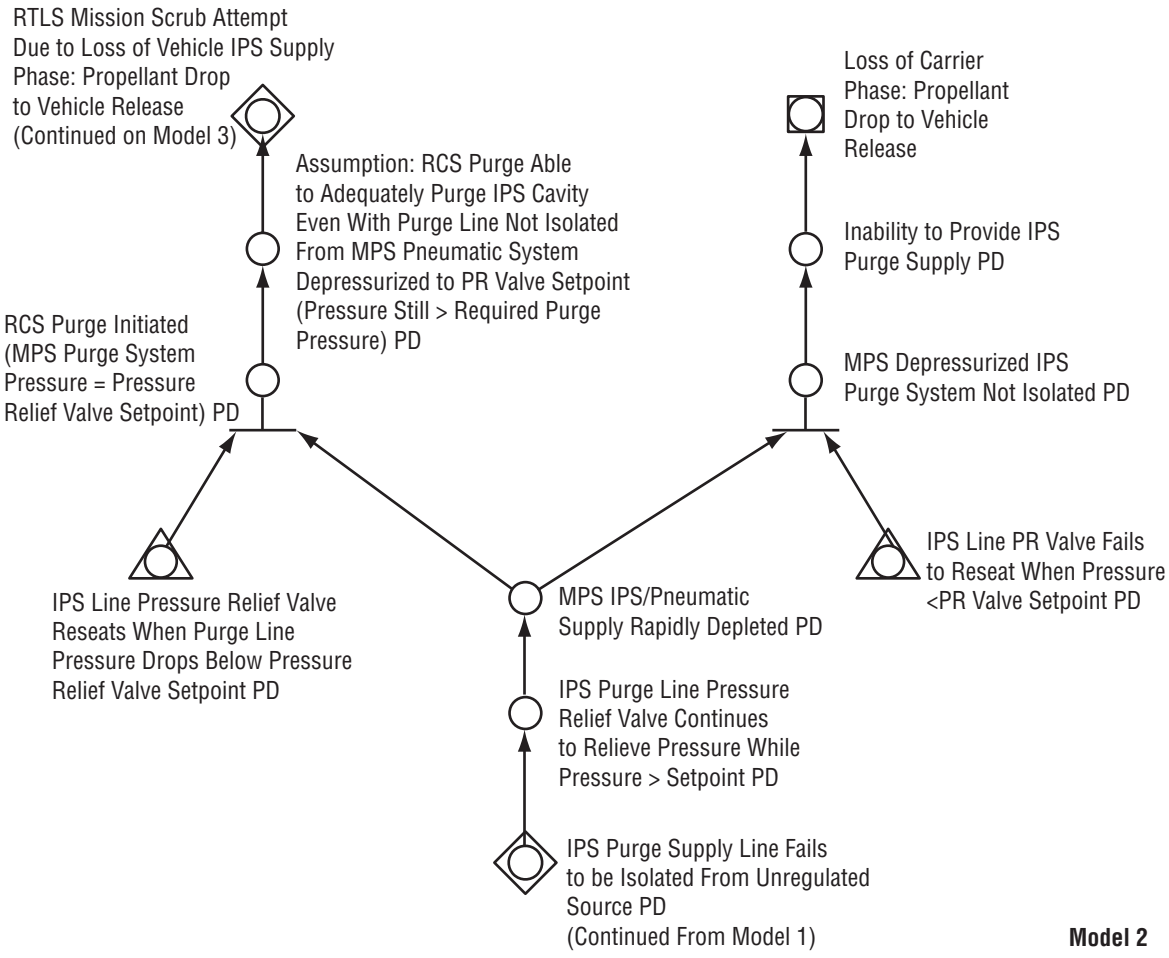


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 2).

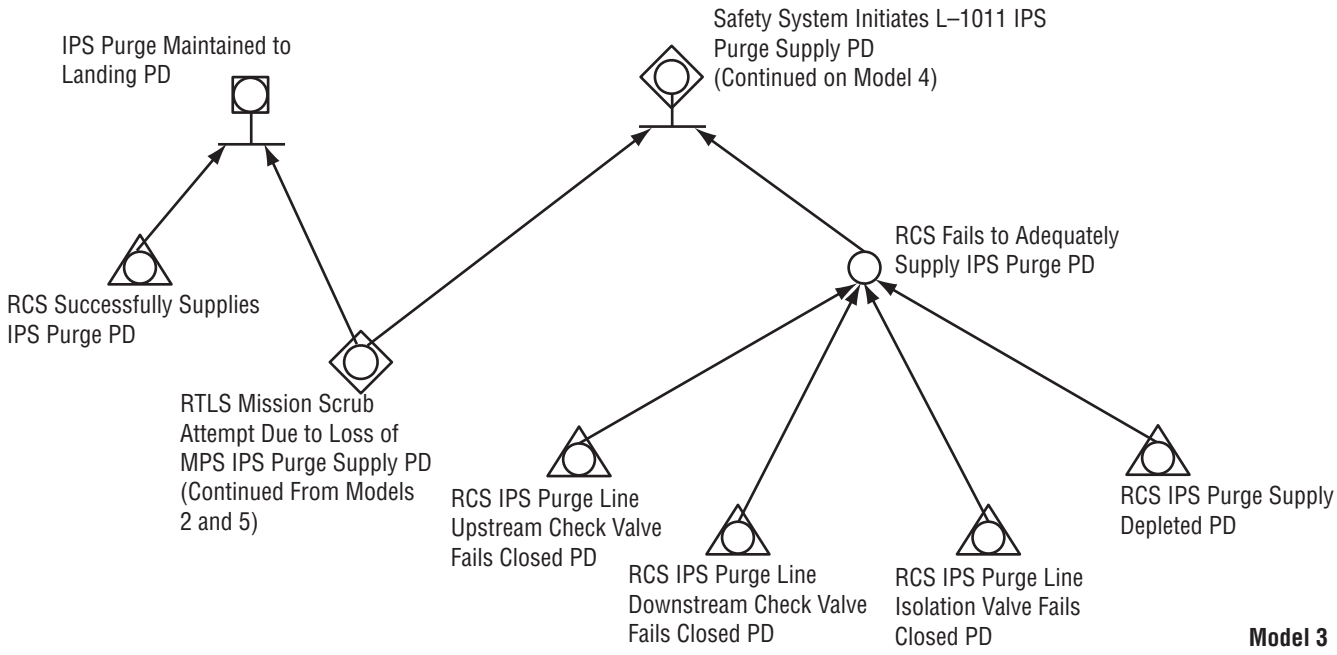


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 3).

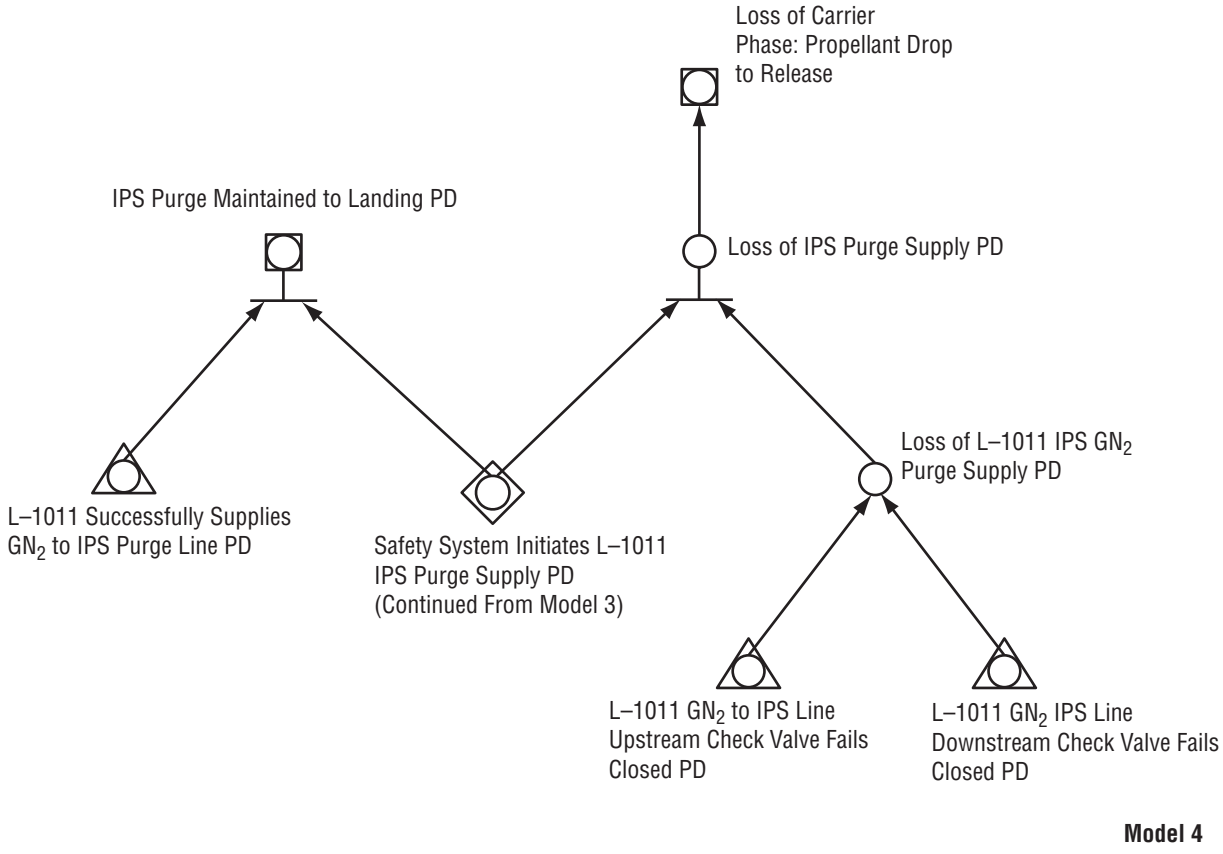


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 4).

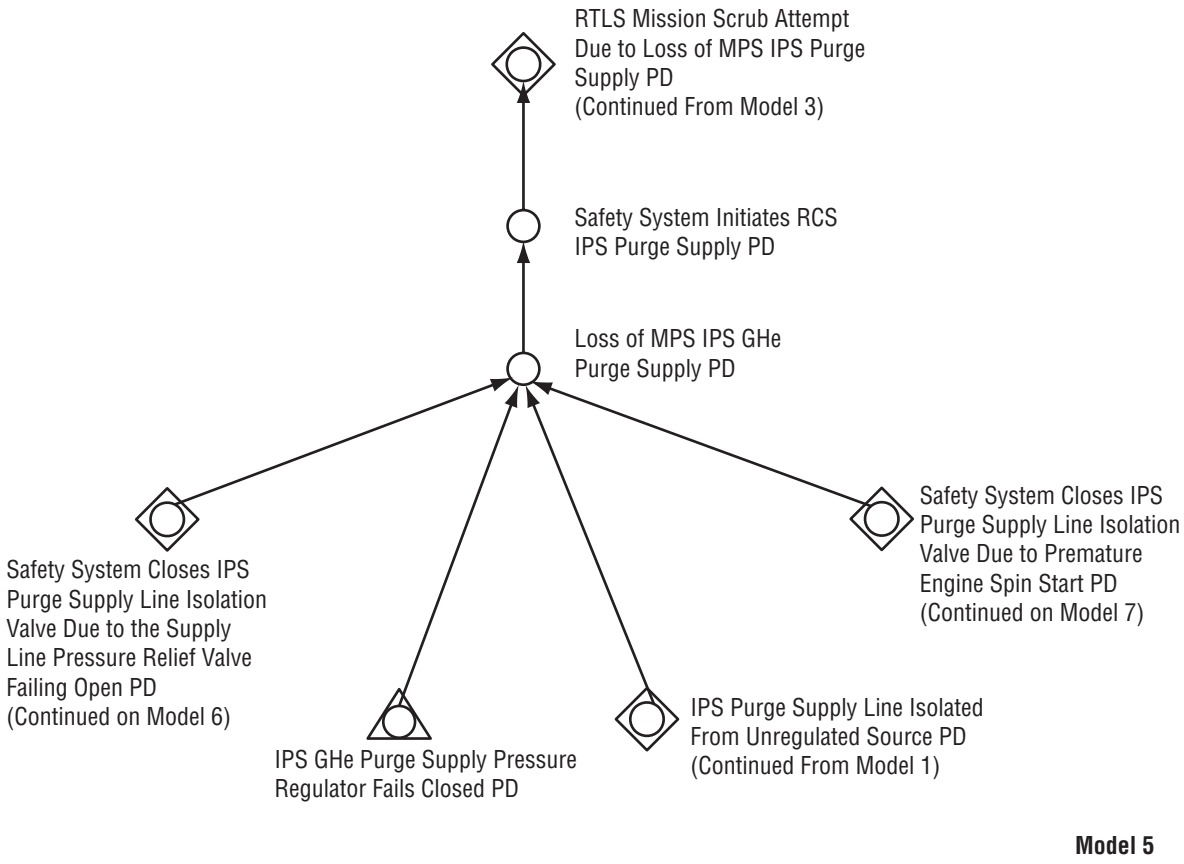


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 5).

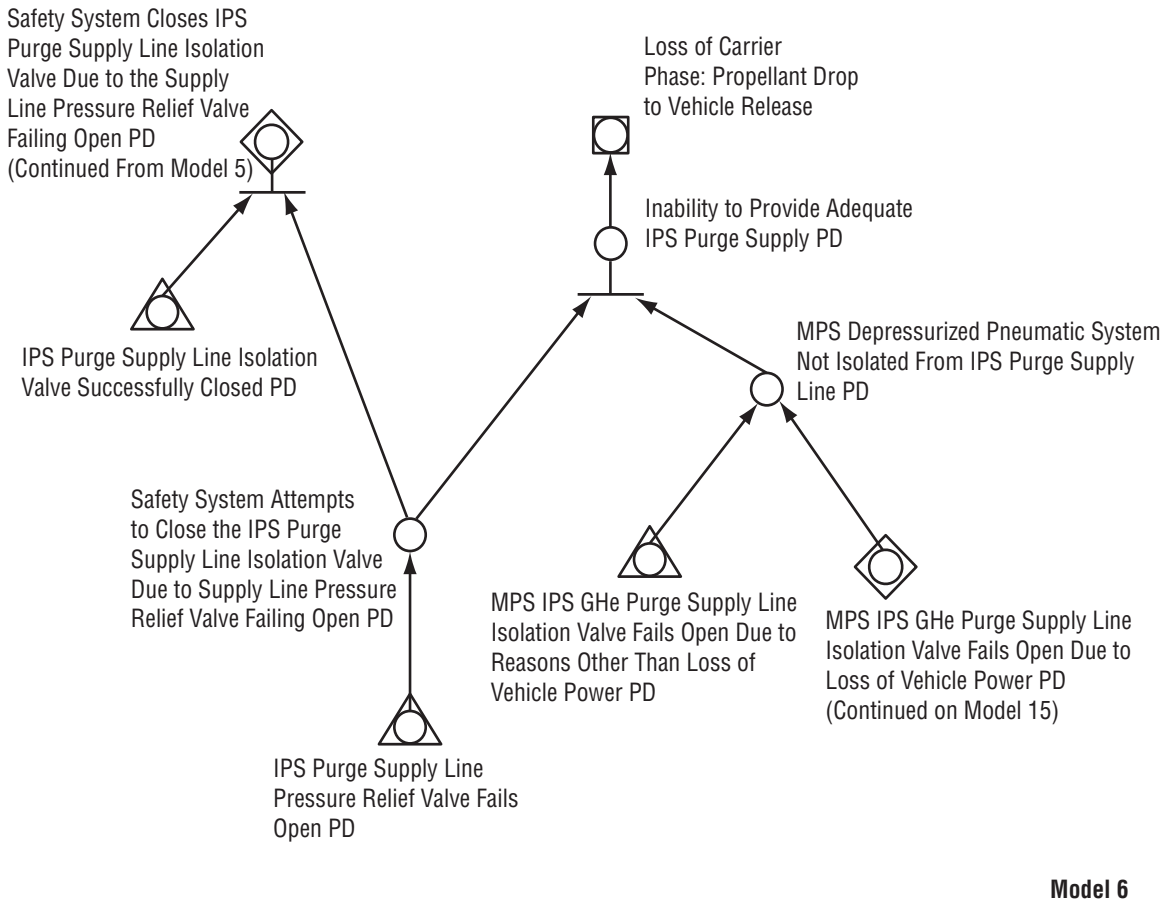


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 6).

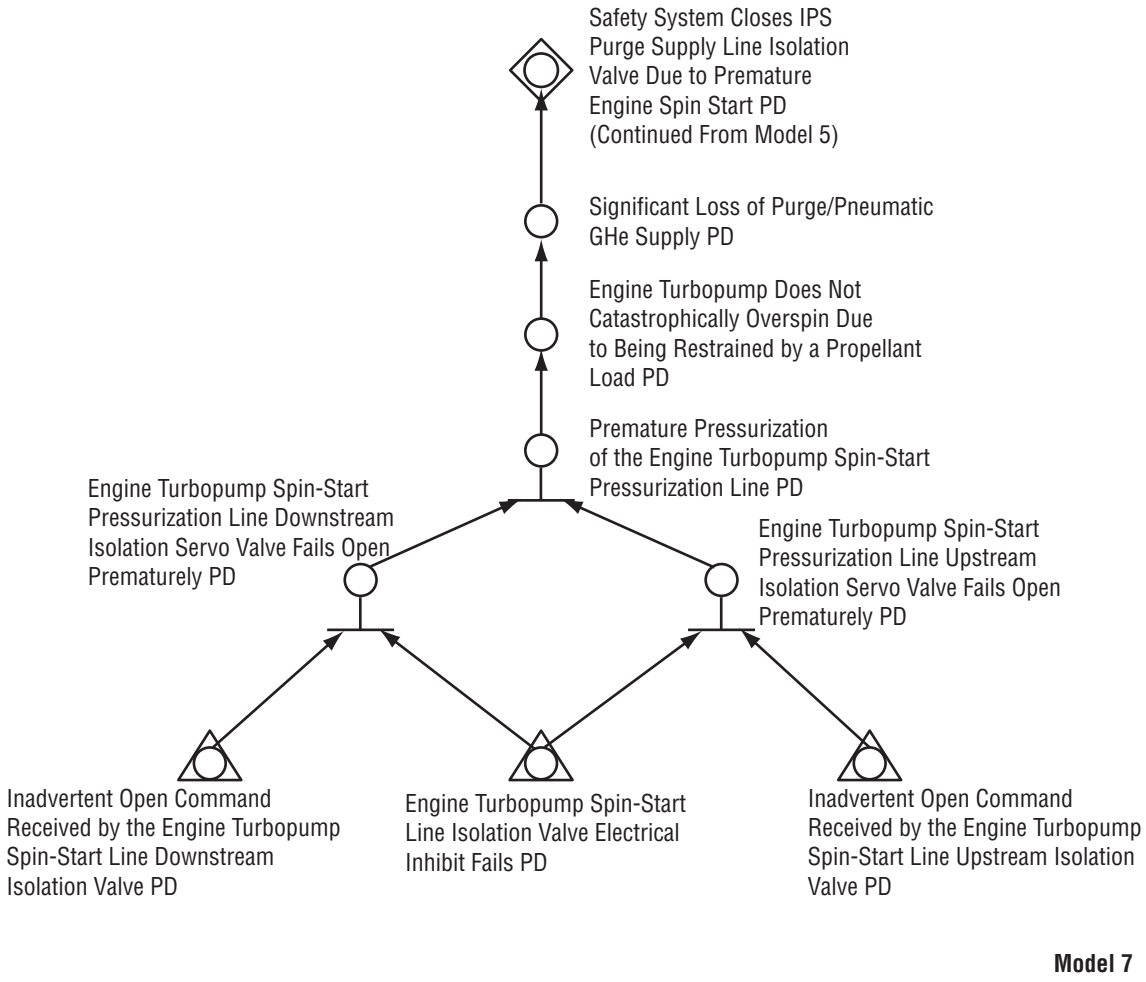


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 7).

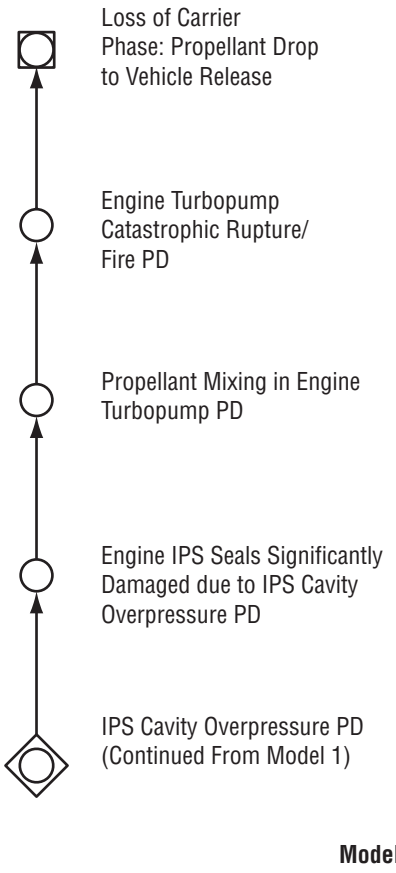


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 8).

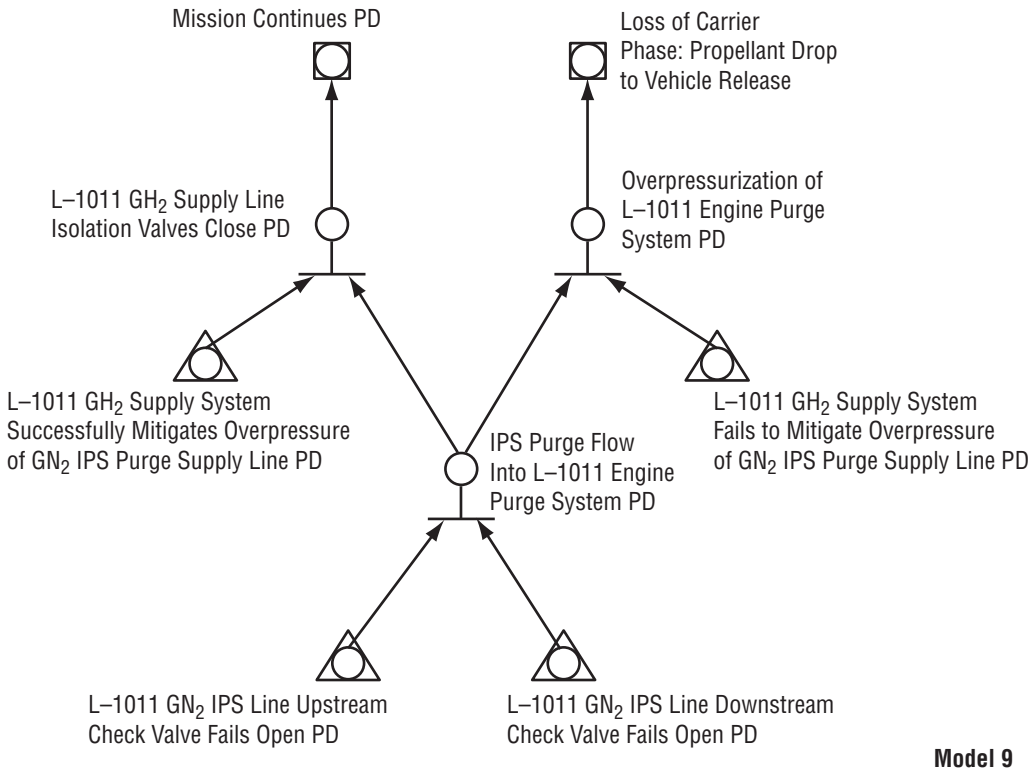


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 9).

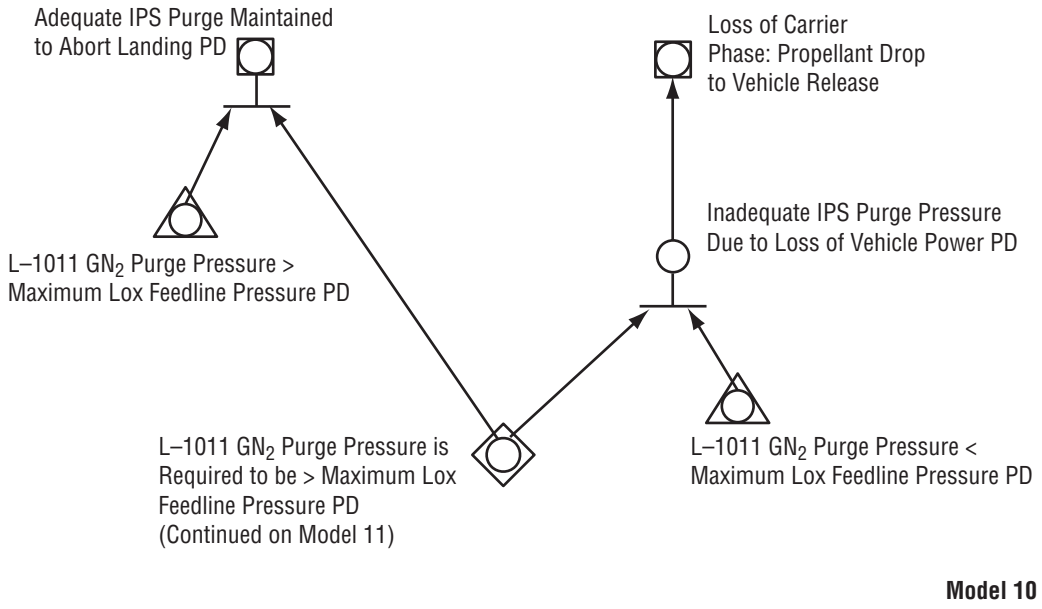


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 10).

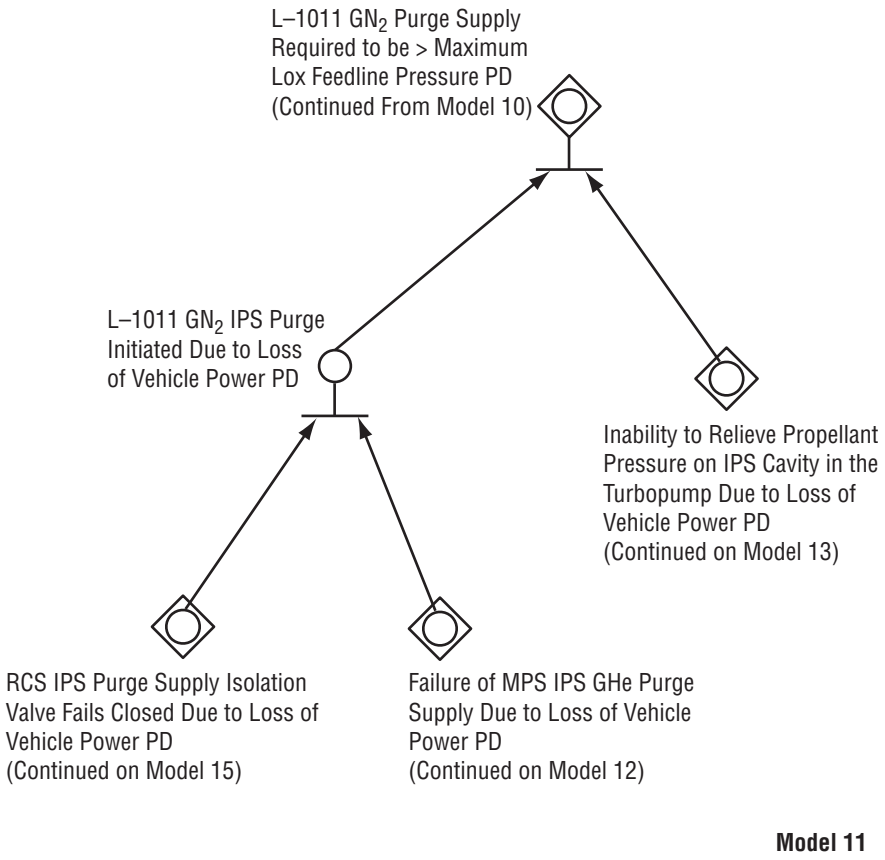


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 11).

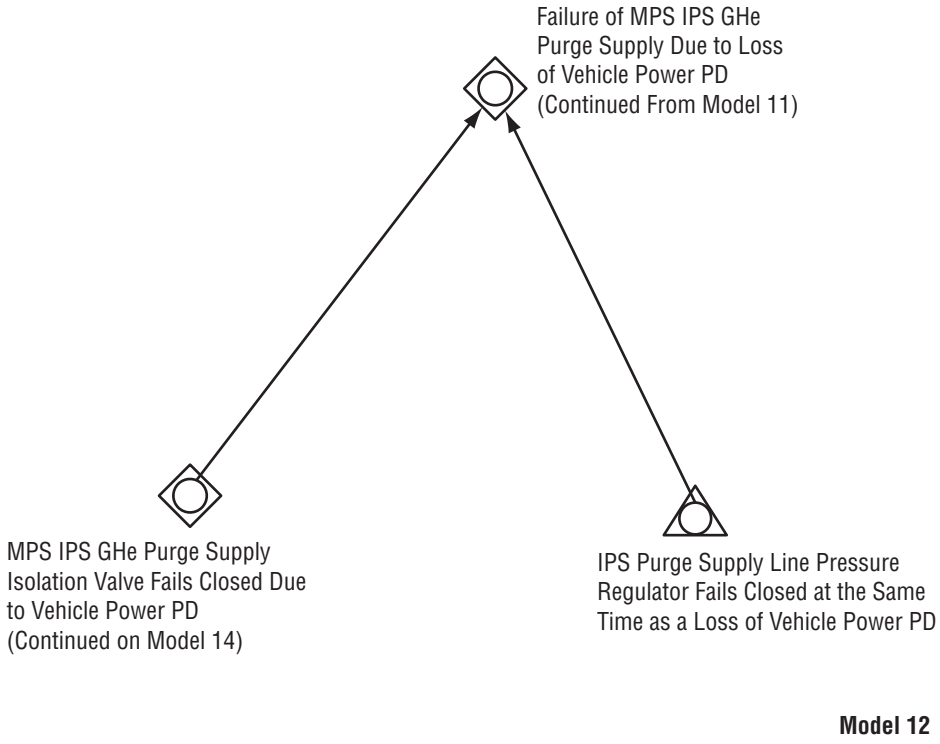


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 12).

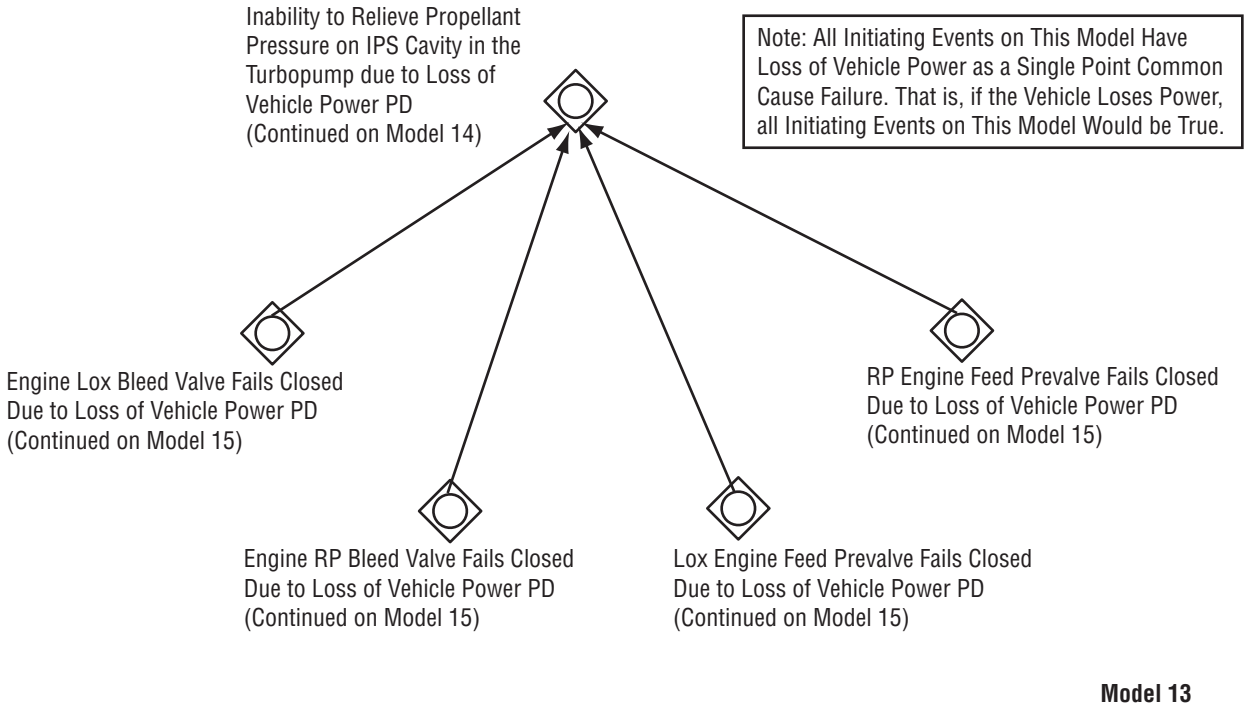


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 13).

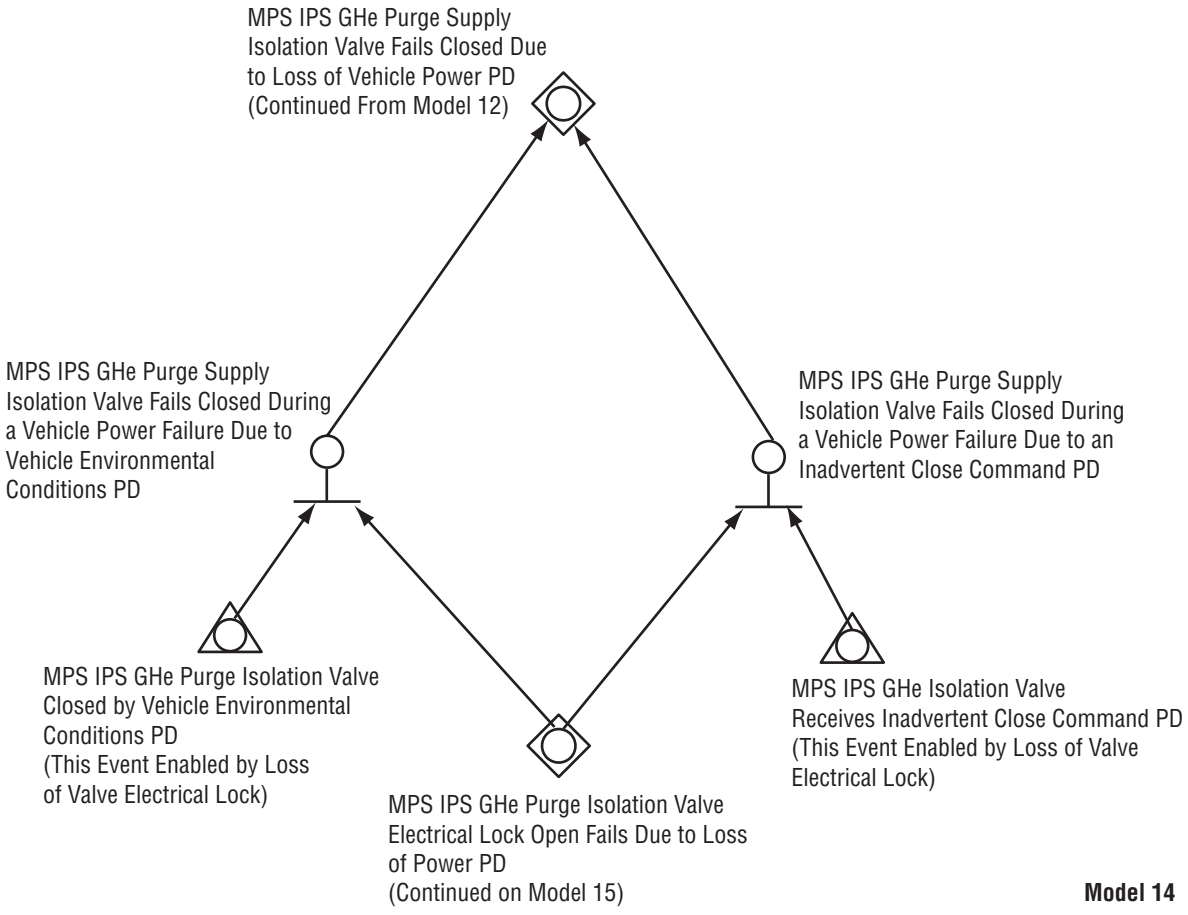


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 14).

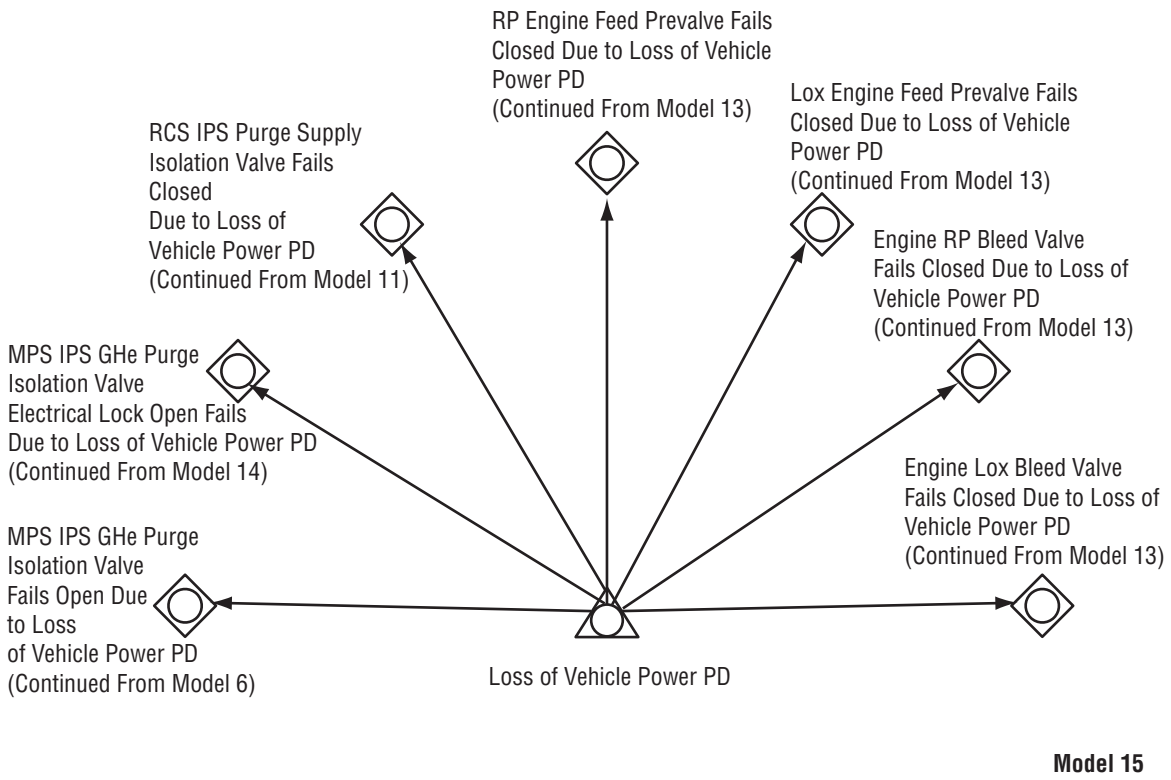


Figure 44. X–34 MPS failure propagation models, pneumatic purge system (Model 15).

The logic models of figure 44 are explicitly linked together through the use of "model" numbers referred to in specific node text. For example, in the top left node description in the first model it says "continue to model 2." The very next figure is "model 2." The model numbers are noted in the lower right-hand corner of each individual logic model.

Assumptions that can be made are:

1. The carrier can safely land with the X–34 vehicle, even if the vehicle has full propellant tanks.

2. Self-pressurization of RP–l during the captive/carry phase does not pose a credible risk.

3. A premature spin-start activation before a propellant load is on the turbopumps results in catastrophic turbopump structural failure.

4. A premature spin-start activation after a propellant load is on the turbopumps will not result in a catastrophic failure.

5. Excessive pneumatic system pressure will cause closed pneumatic valves to fail closed.

6. Excessive pneumatic system pressure will cause structural rupture to open pneumatic valves. However, safety system is assumed to be able to react in time to prevent this failure mode (per MPS team meeting 2/18/98).

7. Excessive purge pressure to the engine IPS cavity results in catastrophic pump failure.

8. Inadvertent dumping or leaking of propellant during captive carry does not pose a significant catastrophic risk (per CDR RID 184, OSC evaluated and approved this assumption pending LN_2 testing during the flight test program).

9. Loss of engine IPS cavity purge while the vehicle is attached to the carrier results in a catastrophic loss of carrier. Therefore, IPS cavity purge must be maintained while the vehicle is attached to the carrier once propellant has been dropped into the engine, including during a mission scrub event where the vehicle is returned with the carrier.

10. Loss of engine IPS cavity purge after vehicle is released from the carrier will cause a turbopump fire/explosion. However, the resulting explosive yield is not sufficient to cause catastrophic damage to the carrier once the vehicle is released from the carrier.

11. The carrier-supplied IPS cavity backup purge is adequate to prevent propellant mixing in the IPS cavity until the vehicle is safely on the ground.

12. Impingement of lox on tank pressurization system servo valves will result in external propellant leakage into MPS bays and fire/explosion.

13. Failure mitigation using closure of the IPS isolation valve (SV11) assumes that closure occurs within 1 sec of the loss of MPS IPS line pressurization (per MSFC turbomachinery analysis). Orbital should ensure that controlling SV11 from the L–1011 LPO station allows SV11 to be closed within 1 sec.

Ground rules:

1. Models are consistent to X–34 Flight Schematic with Instrumentation, Version Xl0.

2. Models incorporated inputs from the technical interchange meeting at Orbital, November 12–14, 1997, and results from CDR RID's.

3. Electrical inhibits to inadvertent valve actuation are considered an acceptable failure-tolerance capability.

4. Catastrophic failures (resulting in loss of life and carrier) were modeled.

5. Single-point structural failures occurring under nominal operating conditions were not modeled.

6. Failures initiated by human error were not modeled.

7. Fill and purge line close-out caps and valves were not included in the model.

8. Four-fault and higher tolerances were not modeled.

APPENDIX D—MPS Quantitative Analysis Support Data

This section provides further quantification data on key MPS components including valves, lines, and ducts (tables 5–9). It follows the conclusions of section 6 to search reliability data sources and use, if possible, direct failure data on operational systems, even if they are surrogate systems. The sources of the data are also described in section 6 with references provided.

Table 5. Solenoid valve failure rate quantification.

Number	Description	Size
V5	He bypass valve (LO_2)	0.5
V7	LO_2 helium tank isolation valve	0.5
V8	LO_2 helium tank fill and drain valve	0.5
V11	GH_2 bypass valve	0.5
V14	He inject isolation valve	0.375

Description	Source	Composite (/hr)	Fail Open (/hr)	Fail Closed (/hr)	Fail to Contain (/hr)
(Prepress solenoid)	SIRA		9.14E–07	9.14E–07	9.20E–05
(Summary, all solenoid operated)	Rome	3.64E–05			
(Composite, all process control valves)	Process Industy		3.00E–07	3.00E–07	1.00E–08
(All Solenoids)	Green & Bourn	3.04307E–05			
(All solenoids)	Anyakora, Engel, & Lees	4.91573E–05			
(All solenoids)	Lawley & Kletz		1.17E–05	3.51E–06	
(All solenoids)	AEC		2.00E–05	2.00E–05	1.00E–08
(Composite, all soleniod, NC)	IEEE	8.52E–05	4.13E–05	4.38E–05	1.00E–07

Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(Prepress Solenoid)	SIRA		1.52E–07	1.52E–07	1.53E–05
(Aircraft)	Rome	6.06E–06			
(Composite, all process control valves)	Process Industy		5.00E–08	5.00E–08	1.67E–09
(All solenoids)	Green & Bourn	5.07E–06			
(All solenoids)	Anyakora, Engel, & Lees	8.19E–06			
(All solenoids)	Lawley & Kletz		1.95E–06	5.85E–07	
(All solenoids)	AEC		3.33E–06	3.33E–06	1.67E–09
(Composite, all soleniod, NC)	IEEE	1.42E–05	6.88E–06	7.30E–06	1.67E–08

Calculate Composites Using "OR" Logic

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(Prepress solenoid)	SIRA	**1.56379E–05**	1.52E–07	1.52E–07	1.53E–05
(Aircraft)	Rome	6.06E–06			
(Composite, all process control valves)	Process Industy	**1.01667E–07**	5.00E–08	5.00E–08	1.67E–09
(All solenoids)	Green & Bourn	5.07E–06			
(All solenoids)	Anyakora, Engel, & Lees	8.19E–06			
(All solenoids)	Lawley & Kletz	**2.53589E–06**	1.95E–06	5.85E–07	
(All solenoids)	AEC	**6.66831E–06**	3.33E–06	3.33E–06	1.67E–09
(Composite, all soleniod, NC)	IEEE	1.42E–05	6.88E–06	7.30E–06	1.67E–08

Calculate Averages and LN Averages Using a Weighting Factor of "1" For All Since They Are Fairly Close
Compare the Resulting Composites and Modes With the "OR" of the Modes

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)	Composite of Modes	DELTA %
(Prepress solenoid)	SIRA	1.56379E–05	1.52E–07	1.52E–07	1.53E–05		
(Aircraft)	Rome	6.06E–06					
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09		
(All solenoids)	Green & Bourn	5.07E–06					
(All solenoids)	Anyakora, Engel, & Lees	8.19E–06					
(All solenoids)	Lawley & Kletz	2.53589E–06	1.95E–06	5.85E–07			
(All solenoids)	AEC	6.66831E–06	3.33E–06	3.33E–06	1.67E–09		
(Composite, all soleniod, NC)	IEEE	1.42E–05	6.88E–06	7.30E–06	1.67E–08		
Averages		**7.30838E–06**	**2.47325E–06**	**2.28416E–06**	**3.8383E–06**	**8.59569E–06**	**–17.61425**
LN Averages		**4.19628E–06**	**8.06274E–07**	**6.41289E–07**	**2.90265E–08**	**1.47659E–06**	**64.811977**

Using the LN Average and Average (Composite of Modes Matches the Actual Composite Best)
Calculate Average of the Composites to Not Over Emphasize the Significance of the Modes or the Actual Coposite
Then Use the Distribution of Modes LN Averages for Distributing This New Composite Number

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)	Composite of Modes	DELTA %
(Prepress solenoid)	SIRA	1.56379E–05	1.52E–07	1.52E-07	1.53E–05		
(Aircraft)	Rome	6.06E–06					
(composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E-08	1.67E–09		
(All solenoids)	Green & Bourn	5.07E–06					
(All solenoids)	Anyakora, Engel, & Lees	8.19E–06					
(All solenoids)	Lawley & Kletz	2.53589E–06	1.95E–06	5.85E–07			
(All solenoids)	AEC	6.66831E–06	3.33E–06	3.33E–06	1.67E–09		
(Composite, all soleniod, NC)	IEEE	1.42E–05	6.88E–06	7.30E–06	1.67E–08		
Averages		7.30838E–06	2.47325E–06	2.28416E–06	3.8383E–06	8.59569E–06	
LN Averages		4.19628E–06	8.06274E–07	6.41289E–07	2.90265E–08	1.47659E–06	–17.614247
New Composite and Modes		**2.83644E–06**	**1.5488E–06**	**1.23188E–06**	**5.57581E–08**	**2.83644E–06**	**64.8119767**

These Probabilities Can Then be Converted Back to Time-to-Failure Exponential Distributions and to Reliabilities

New Composite and Modes	2.83644E–06	1.5488E–06	1.23188E–06	5.57581E–08
LAMBDA (SEC)	**4.7274E–09**	**2.58134E–09**	**2.05313E–09**	**9.29302E–11**
Reliability	**0.999997164**	**0.999998451**	**0.999998768**	**0.999999944**

Table 6. Relief valve failure rate quantification.

Number	Description	Size
V2	LO_2 relief valve	5
V9	LH_2 relief valve	5

Description	Source	Composite (/hr)	Fail Open (/hr)	Fail Closed (/hr)	Fail to Contain (/hr)
(He relief)	SIRA			1.13E–07	9.20E–05
(Composite, all pneumatic reliefs)	Rome	7.80E–06			
(Composite, all process control valves)	Process Industy		3.00E–07	3.00E–07	1.00E–08
(All reliefs)	Green & Bourn	2.57491E–06			
(All reliefs)	Green & Bourn–2		2.00E–06	5.00E–07	
(All reliefs)	Lawley & Kletz		2.00E-06	5.00E–07	
(All reliefs)	AEC		1.00E–05	4.20E–07	
(Composite, all pressure relief)	IEEE	5.00E–06			

Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(He relief)	SIRA			1.88E–08	1.53E–05
(Composite, all pneumatic reliefs)	Rome	1.30E–06			
(Composite, all process control valves)	Process Industy		5.00E–08	5.00E–08	1.67E–09
(All reliefs)	Green & Bourn	4.29E–07			
(All reliefs)	Green & Bourn–2		3.33E–07	8.33E–08	
(All reliefs)	Lawley & Kletz		3.33E–07	8.33E–08	
(All reliefs)	AEC		1.67E–06	7.00E–08	
(Composite, all pressure relief)	IEEE	8.33E–07			

Calculate Composites Using "OR" Logic

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(He relief)	SIRA	**1.5352E–05**		1.88E–08	1.53E–05
(Composite, all pneumatic reliefs)	Rome	1.30E–06			
(Composite, all process control valves)	Process Industy	**1.01667E–07**	5.00E–08	5.00E–08	1.67E–09
(All reliefs)	Green & Bourn	4.29E–07			
(All reliefs)	Green & Bourn–2	**4.16667E–07**	3.33E–07	8.33E–08	
(All reliefs)	Lawley & Kletz	**4.16667E–07**	3.33E–07	8.33E–08	
(All reliefs)	AEC	**1.73667E–06**	1.67E–06	7.00E-08	
(Composite, all pressure relief)	IEEE	8.33E–07			

Calculate Averages and LN Averages Using a Weighting Factor OF "1" For all Since They are Fairly Close
Compare the Resulting Composites and Modes With the "OR" of the Modes

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)	Composite of Modes	Delta %
(He relief)	SIRA	1.5352E–05		1.88E–08	1.53E–05		
(Composite, all pneumatic reliefs)	Rome	1.30E–06					
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09		
(All reliefs)	Green & Bourn	4.29E–07					
(All reliefs)	Green & Bourn-2	4.17E–07	3.33E–07	8.33E–08			
(All reliefs)	Lawley & Kletz	4.16667E–07	3.33E–07	8.33E–08			
(All reliefs)	AEC	1.73667E–06	1.67E–06	7.00E–08			
(Composite, all pressure relief)	IEEE	8.33E–07					
Averages		**2.57327E–06**	**5.95833E–07**	**6.11E–08**	**7.66744E–06**	**8.32437E–06**	**–223.4932276**
LN Averages		**8.26992E–07**	**3.10202E–07**	**5.39668E–08**	**1.5986E–07**	**5.24029E–07**	**36.63436469**

Using the LN Average (Average Deviates to FAR) (Composite of Modes Matches the Actual Composite Best)
Then use the Distribution of Modes LN Averages for Distributing This Composite Number

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)	Composite of Modes	Delta %
(Prepress solenoid)	SIRA	1.5352E–05	0.00E+00	1.88E–08	1.53E–05		
(Aircraft)	Rome	1.30E–06					
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09		
(All solenoids)	Green & Bourn	4.29E–07					
(All solenoids)	Anyakora, Engel, & Lees	4.17E–07					
(All solenoids)	Lawley & Kletz	4.16667E–07	3.33E–07	8.33E–08			
(Composite, all pressure relief)	IEEE	8.33E–07	0.00E+00	0.00E+00	0.00E+00		
Averages		2.57327E–06	5.95833E–07	6.11E–08	7.66744E–06	8.32437E–06	
LN Averages		8.26992E–07	3.10202E–07	5.39668E–08	1.5986E–07	5.24029E–07	–223.4932276
New Composite and Modes		**6.7551E–07**	**3.99872E–07**	**6.9567E–08**	**2.06071E–07**	**6.7551E–07**	**36.63436469**

These Probabilities can Then be Converted Back to Time-to-Failure Exponential Distributions and to Reliabilities

New Composite and Modes	6.7551E–07	3.99872E–07	6.9567E–08	2.06071E–07
LAMBDA (SEC)	**1.12585E–09**	**6.66453E–10**	**1.15945E–10**	**3.43453E–10**
Reliability	**0.999999324**	**0.9999996**	**0.99999993**	**0.999999794**

Table 7. Check valve failure rate quantification.

Number	Description	Size
V6	He prepress check valve (LO_2)	1
V12	GH_2 out-press check valve	1.6
V13	He prepress check valve (LH_2)	1
V15	By-Pass pilot check valve	0.5

Description	Source	Composite (/hr)	Fail Open (/hr)	Fail Closed (/hr)	Fail to Contain (/hr)
(He Pneumatic Check)	SIRA		7.45E–06	1.44E–06	9.20E–05
(Composite, all check valves)	Rome	7.80E–06			
(Composite, all process control valves)	Process Industy		3.00E–07	3.00E–07	1.00E–08
(Composite, all checks valves)	AEC		3.00E–07	4.00E–04	1.00E–08
(Composite, all pressure relief)	IEEE		5.00E–07	3.00E–07	5.00E–08

Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(He pneumatic check)	SIRA		1.24E–06	2.40E–07	1.53E–05
(Composite, all check valves)	Rome	1.30E–06			
(Composite, all process control valves)	Process Industy		5.00E–08	5.00E–08	1.67E–09
(Composite, all checks valves)	AEC		5.00E–08	6.67E–05	1.67E–09
(Composite, all pressure relief)	IEEE		8.33E–08	5.00E–08	8.33E–09

Calculate Composites Using "OR" Logic

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)
(He pneumatic check)	SIRA	**1.68149E–05**	1.24E–06	2.40E–07	1.53E–05
(Composite, all check valves)	Rome	1.30E–06			
(Composite, all process control valves)	Process Industy	**1.01667E–07**	5.00E–08	5.00E–08	1.67E–09
(Composite, all checks valves)	AEC	**6.67161E–05**	5.00E–08	6.67E–05	1.67E–09
(Composite, all pressure relief)	IEEE	**1.41667E–07**	8.33E–08	5.00E–08	8.33E–09

Calculate Averages and LN Averages Using a Weighting Factor of "1" for all Since They are Fairly Close
Compare the Resulting Composites and Modes with the "OR" of the Modes

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)	**Composite of Modes**	**Delta %**
(He Pneumatic Check)	SIRA	1.68149E–05	1.24E–06	2.40E–07	1.53E–05		
(Composite, all check valves)	Rome	1.30E–06					
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09		
(Composite, all checks valves)	AEC	6.67E–05	5.00E–08	6.67E–05	1.67E–09		
(Composite, all pressure relief)	IEEE	1.42E–07	8.33E–08	5.00E–08	8.33E–09		
AVERAGES		**1.70149E–05**	**3.5625E–07**	**1.67511E–05**	**3.83622E–06**	**2.09435E–05**	**–23.08952427**
LN AVERAGES		**1.8385E–06**	**1.26821E–07**	**4.4721E–07**	**2.44083E–08**	**5.98439E–07**	**67.44953213**

Using the LN Average (Average Deviates to FAR) (Composite of Modes Matches the Actual Composite Best)
Then use the Distribution of Modes LN Averages for Distributing this Composite Number

		Composite (*P* fail)	Fail Open (*P* fail)	Fail Closed (*P* fail)	Fail to Contain (*P* fail)	**Composite of Modes**	**Delta %**
(He Pneumatic Check)	SIRA	1.68149E–05	1.24E–06	2.40E–07	1.53E–05		
(Composite, all check valves)	Rome	1.30E–06					
(Composite, all process control valves)	Process Industy	1.01667E–07	5.00E–08	5.00E–08	1.67E–09		
(Composite, all checks valves)	AEC	6.67E–05	5.00E–08	6.67E–05	1.67E–09		
(Composite, all pressure relief)	IEEE	1.42E–07	8.33E–08	5.00E–08	8.33E–09		
Averages		1.70149E–05	3.5625E–07	1.67511E–05	3.83622E–06	2.09435E–05	
LN Averages		1.8385E–06	1.26821E–07	4.4721E–07	2.44083E–08	5.98439E–07	–23.08952427
New Composite and Modes		**1.21847E–06**	**2.58217E–07**	**9.10553E–07**	**4.96971E–08**	**1.21847E–06**	**67.44953213**

These Probabilities Can Then be Converted Back to Time-to-Failure Exponential Distributions and to Reliabilities

New Composites and Modes	1.21847E–06	2.58217E–07	9.10553E–07	4.96971E–08
LAMBDA (sec)	**2.03078E–09**	**4.30362E–10**	**1.51759E–09**	**8.28286E–11**
Reliability	**0.999998782**	**0.999999742**	**0.999999089**	**0.99999995**

Table 8. Feedline failure rate.

Number	Description	Size
L1	LO_2 feed, fill, and drain	4
L2	LH_2 feed, fill and drain	4
L3	GO_2 press, vent, and relief lines	1
L4	GH_2 press, vent, and relief IIne s	1

Description	Source	Composite	Pipe	Flanges	Welds
(SIRA, Weld Risk, 4 welds)	SIRA	1.50E–06			
(Ducts)	Rome	2.10E–05			
(All pipe ≤ 3-in. section, 2 flanges, 4 welds)	Process Industy	6.0013E–06	1.00E–10	6.00E–06	1.20E–09

Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions

		Composite (*P* fail)
(SIRA, weld risk, 4 welds)	SIRA	2.50E–07
(Ducts)	Rome	3.50E–06
(All pipe ≤ 3-in. section, 2 flanges, 4 welds)	Process Industy	1.00E–06

Calculate Averages and LN Averages Using a Weighting Factor of "1" for all Since They are Fairly Close

		Composite (*P* fail)
(SIRA, Weld Risk, 4 welds)	SIRA	2.50E–07
(Ducts)	Rome	3.50E–06
(All pipe ≤ 3-in. section, 2 flanges, 4 welds)	Process Industy	1.00E–06
Averages		**1.5834E–06**
LN Averages		**9.56534E–07**

Using LN Average, These Probabilities Can Then be Converted Back to Time-to-Failure Exponential Distributions and to Reliabilities

LAMBDA (sec)	**1.59422E–09**
Reliability	**0.999999043**

Table 9. Duct failure rate quantification.

Number	Description	Size
B1		8
B2	Bellows	6
B3	Bellows,LH_2	4
B4	Bellows, GO_2	4
B5	Bellows, GH_2	4
B6	BSTRA	8
B7	BSTRA	4
B8	BSTRA, GO_2	2
B9	BSTRA. GH_2	2

Description	**Source**	**Composite**
(Bellows)	Rome	8.27E–08
(Expansion Joints)	Process Industy	3.00E–07

Calculate Probabilities Assuming a 600-Sec Mission and Exponential Distributions

		Composite (*P* fail)
(Ducts)	Rome	1.38E–08
(All pipe ≤ 3-in. section, 2 flanges, 4 welds)	Process Industy	5.00E–08

Calculate Averages and LN Averages Using a Weighting Factor of "1" for all Since They are Fairly Close

		Composite (*P* fail)
(Ducts)	Rome	1.38E–08
(All pipe ≤ 3-in. section, 2 flanges, 4 welds)	Process Industy	5.00E–08
Averages		**3.18917E–08**
LN Averages		**2.6252E–08**

Using LN Average, These Probabilities can Then be Converted Back to Time to Failure Exponential Distributions and to Reliabilities

LAMBDA (sec)	**4.37533E–11**
Reliability	**0.999999974**

REFERENCES

1. Pye, D.: *The Nature of Design,* Reinhold Book Corp., New York, NY 1969.

2. Petroski, H.: *To Engineer is Human,* St. Martin Press, New York, NY 1982.

3. Ryan, R.S.; et al.: "Working on the Boundaries: Philosophies and Practices of the Design Process," NASA–TP–3642, MSFC, July 1996.

4. Ryan, R.S.; and Verderaime, V.: "Systems Design Analysis Applied to Launch Vehicle Configuration," NASA–TP–3326, MSFC, January 1993.

5. McCarty, J.P.: "A Critical Function Technique for Modeling Launch Vehicle Reliability," *A Dissertation,* University of Alabama in Huntsville, Huntsville, AL, 1996.

6. "Reliability Design and Verification for Launch Vehicle Propulsion Systems," AIAA Workshop on Reliability of Launch Vehicles Propulsion Systems, Washington, DC, May 1989.

7. Fragola, J.R.: *A Second Look at Launch System Reliability,* Aerospace America, pp. 36–39, November 1991.

8. McFadden, R.H.; and Shen, Y.: "An Analysis of the Historical Reliability of US Liquid-Fuel Propulsion Systems," *AIAA–90–2713,* 26th Joint Propulsion Conference, Orlando, FL, July 1990.

9. Christenson, R.L.; and Komar, D.R.: "Reusable Rocket Engine Operability Modeling and Analysis," *NASA–TP–208530,* MSFC, July 1998.

10. Fragola, J.R.: "Risk Management in US Manned Spacecraft: From Apollo to Alpha and Beyond," *Proceedings of the ESA 1996 Product Assurance Symposium and Software Product Assurance Workshop,* Norordwijk, The Netherlands, March 1996.

11. "Procedures for Performing a Failure Mode, Effects, and Criticality Analysis," *MIL–STD–1629A,* November 1980.

12. Goldberg, B.E.; et al.: "System Engineering "Toolbox" for Design-Oriented Engineers," *NASA Reference Publication 1358,* December 1994.

13. "Reliability Program for Systems and Equipment Development and Production," *MIL–STD–785B,* September 1980.

14. "Reliability Program Requirements for Space Launch Vehicles," *MIL–STD–1543B*, October 1988.

15. *MIL–STD–756B,* "Reliability Modeling and Prediction," November 1981.

16. Kapur, K.C.; and Lamberson, L.R.: *Reliability in Engineering Design,* Wiley, New York, NY 1977.

17. Knight, K.: *FEAS–M User's Guide,* Sverdrup Technology, July 1998.

18. Ryan, R.S.; and Townsend, J.: "Application of Probabilistic Analysis/Design Methods in Space Programs," *AIAA Journal of Spacecraft and Rockets,* Vol.31, No.6, pp.1038–1043, November 1994.

19. Wirsching, P.H.: Lecture Notes on Study Course Entitled "Reliability Methods in Mechanical and Structural Design," MSFC, May 1992.

20. Safie, F.; and Fox, E.P.: "A Probabilistic Design Analysis Approach for Launch Systems," *AIAA–91–3372, 27th Joint Propulsion Conference,* Sacramento, CA, June 1991.

21. "Probabilistic Structures Analysis Methods for Select Space Propulsion System Components (PSAM)," Southwest Research Institute, NASA–CR–NAS3–24389, March 1986.

22. "Probabilistic Risk Assessment of the Space Shuttle Phase I: Space Shuttle Catastrophic Failure Frequency Final Report," Science Applications International Corporation, NASA Contract No. NAS6–25809, 1993.

23. "Independent Assessment of Shuttle Accident Scenario Probabilities for Galileo Mission and Comparison with NSTS Program Assessment," Planning Research Corporation, 1989.

24. "1997 Space Shuttle Quantitative Risk Assessment," NASA MSFC, 1997

25. "Shuttle Integrated Risk Assessment," Rockwell International, Space Systems Division, NASA Contract NAS9–18500, September 1990.

26. Moore, N.; et al.: "An Improved Approach for Flight Readiness Certification—Methodology for Failure Risk Assessment and Application," *JPL Publication 92–15,* California Institute of Technology, Pasadena, CA, June 1995.

27. "SAIC Says Shuttle Launch Failure Risk Much Lower After Redesign," *Aerospace Daily,* Vol.175, No. 32, p. 249, August 17, 1995.

28. Townsend, J; et al.: "Review of the Probabilistic Failure Analysis Methodology and Other Probabilistic Approaches for Application in Aerospace Structural Design," *NASA–TP–3434,* MSFC, November 1993.

29. Ebrahimi, N.: "Multistate Reliability Models," *Naval Research Logistics Quarterly,* Vol. 31, pp. 671–680, 1984.

30. Fu, J.C.: "Reliability of Consecutive-k-out-of-n: F Systems with (k–1)-step Markov Dependence," *IEEE Transactions on Reliability,* Vol. R–35, pp. 606–692, 1986.

31. "Reliability Prediction of Electronic Equipment," *MIL–HDBK–217F,* July 1992.

32. Dhillon, B.S.: "Mechanical Reliability: Theory, Models, and Applications," *AIAA*, Washington, DC, 1988.

33. "STS PRACA Database," NASA KSC, l975–current.

34. "Reliability Data for Pumps and Drives, Valve Actuators, and Valves," *ANSI/IEEE Std. 500,* 1984.

35. Brown, R.: "Historical Launch Vehicle Database," MSFC/PD, 1992.

36. Lees, F.P.: *Loss Prevention in the Process Industry,* Butterworths, London, 1980.

37. "Failure Mode/Mechanism Distribution," *FMD–91/97,* Rome Air Development Center.

38. Swain, A.; and Guttman, W.: "Human Reliability," *NUREG,* 1984.

39. Biggs, R.: "SSME Flight Reliability Study Update," *Rocketdyne Rpt. ILCPE–SP–525–91–001,* April 1991.

40. McCormick, E.J.; *Human Factors Engineering,* McGraw-Hill, New York, NY, 1970.

41. Clemens, R.L.; and Mohr, R.R.: Lecture Notes on Study Course Entitled "Risk Management and System Safety Practice," Sverdrup Technology, Huntsville, AL, October 1991.

42. "Standard Practice for System Safety Program Requirements," *MIL–STD–882C,* October 1993.

43. Lloyd, D.K.; and Lipow, M.: "Reliability: Management, Methods, and Mathematics," *The American Society for Quality Control,* Milwaukee, WI, 1984.

44. Lishman, S.: "Unsatisfactory Condition Reports (UCR's) versus Engine Failure," *RLV Task Report,* November 1998.

45. "X–34 Main Propulsion System Specification," Orbital Sciences Corporation: *Ref Doc. X60025,* rev. B, August 1997.

46. "X–34 Integrated Risk Assessment," Safety and Mission Assurance Office, Contract NAS8–40364, MSFC, August 1998.

47. "SSME Automated Configurated-Data Tracking System (TRACER)," Rocketdyne, 1975–current.

48. "Metallic Materials and Elements for Aerospace Vehicle Structures," *MIL–HDBK–5F,* October 1993.

49. "Sampling Procedures and Tables for Inspection by Attributes," *MIL–STD–105E,* October 1993.

50. "Sampling Procedures and Tables for Inspection by Variables for Percent Defective," *MIL–STD–414,* October 1993.

51. Snedecor, G.W.; and Cochran, W.G.: *Statistical Methods,* Iowa State University Press, Ames, IA, 1980.

52. "Reliability Growth Management," *MIL–HDBK–189,* February 1981.

53. Tanija, VS.; and Safie, F.M.: "An Overview of Reliability Growth Models and Their Potential Use for NASA Applications," *NASA–TP–3309,* MSFC, 1992.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	January 2000	Technical Publication

4. TITLE AND SUBTITLE
Comprehensive Design Reliability Activities for Aerospace Propulsion Systems

5. FUNDING NUMBERS

6. AUTHORS
R.L. Christenson, M.R. Whitley, and K.C. Knight*

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)
George C. Marshall Space Flight Center
Marshall Space Flight Center, AL 35812

8. PERFORMING ORGANIZATION REPORT NUMBER
M–958

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Washington, DC 20546–0001

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA/TP—2000–209902

11. SUPPLEMENTARY NOTES
Prepared by Advanced Concepts Department, Space Transportation Directorate
*Sverdrup Technology, Huntsville, Alabama

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified-Unlimited
Subject Category 15
Standard Distribution

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This technical publication describes the methodology, model, software tool, input data, and analysis results that support aerospace design reliability studies. The focus of these activities is on propulsion systems mechanical design reliability. The goal of these activities is to support design from a reliability perspective. Paralleling performance analyses in schedule and method, this requires the proper use of metrics in a validated reliability model useful for design, sensitivity, and trade studies. Design reliability analysis in this view is one of several critical design functions.

A design reliability method is detailed and two example analyses are provided—one qualitative and the other quantitative. The use of aerospace and commercial data sources for quantification is discussed and sources listed. A tool that was developed to support both types of analyses is presented. Finally, special topics discussed include the development of design criteria, issues of reliability quantification, quality control, and reliability verification.

14. SUBJECT TERMS
reliability, propulsion systems, mechanical systems, mechanical reliability, reliability analysis, design criteria, reliability verification, FEAS–M, quality control, design reliability, UCR, MPS, failure rate

15. NUMBER OF PAGES
144

16. PRICE CODE
A07

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited